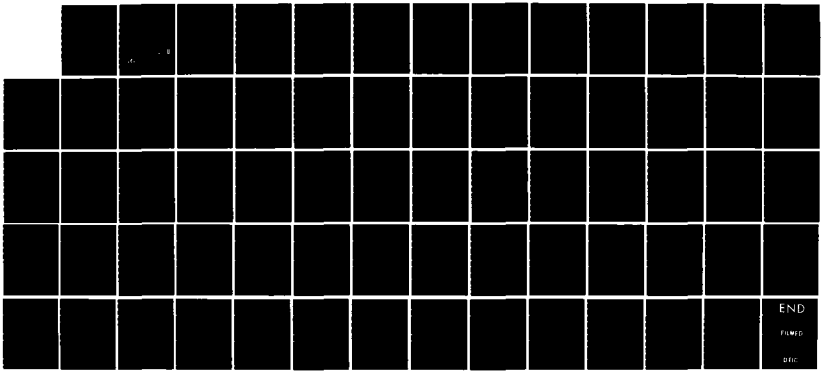
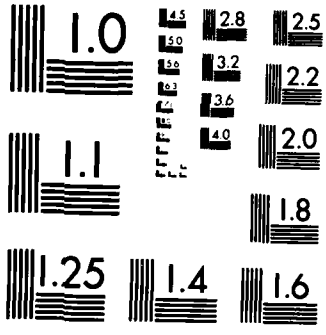
AD-A154 631 PROGRAMS FOR COLOR PLOTTING(U) NAVAL SURFACE WEAPONS CENTER SILVER SPRING MD J M PHIN SEP 83 NSWC/MP-83-392 1/1

UNCLASSIFIED F/G 9/2 NL





NSWC MP 83-392

AD-A154 631

PROGRAMS FOR COLOR PLOTTING

BY J. MARSHAL PHIN

RESEARCH AND TECHNOLOGY DEPARTMENT

SEPTEMBER 1983

NAVAL SURFACE WEAPONS CENTER

Dahlgren, Virginia 22448 • Silver Spring, Maryland 20910

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: NSWC MP 83-392	2. GOVT ACCESSION NO.: A154 681
	3. RECIPIENT'S CATALOG NUMBER:
4. TITLE *(and Subtitle)*: PROGRAMS FOR COLOR PLOTTING	5. TYPE OF REPORT & PERIOD COVERED: Final; Fiscal Year 83
	6. PERFORMING ORG. REPORT NUMBER:
7. AUTHOR(s): J. Marshal Phin	8. CONTRACT OR GRANT NUMBER(s):
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Naval Surface Weapons Center, White Oak (Code R41) Silver Spring, Maryland 20910	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS:
11. CONTROLLING OFFICE NAME AND ADDRESS:	12. REPORT DATE: September 1983
	13. NUMBER OF PAGES: 68
14. MONITORING AGENCY NAME & ADDRESS*(if different from Controlling Office)*:	15. SECURITY CLASS. *(of this report)*: UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE:

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

Computer Program
Digital Plotter
Color

Accession For	
NTIS GRA&I	X
DTIC TAB	☐
Unannounced	☐
~~Justification~~	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A/1	

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

This document explains in detail a group of utility and special purpose computer programs. The programs were written to be used with a micro-computer, with a serial port, and to drive a digital plotter. There are programs in this document that can be used to plot paired laboratory data and functions. There are also programs to draw ellipses, regular polygons, circles and pie graphs. There are two polar plotting programs. One polar program to plot data and another to plot a polar function.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

The programs were written specifically to drive a Houston Hiplot digital plotter with a serial interface. The plotter is a model DPM-43 designed to hold standard typing paper. The plotter is also capable of plotting in six different colors.

The programs in this document use plotting software commands specific to the Hiplot plotter and were written in Northstar Basic.

This document will describe, discuss and list these programs and provide examples of the drawings produced. These programs are interactive and user friendly.

FOREWORD

This report lists basic programs for plotting mathematical functions and experimental data. Menues and automatic file handling help the user create graphic displays that meet general laboratory needs. More specialized routines display the distribution of radiation intensity around an anisotropic point source. Naval Sea Systems Command, Code 644, supported this work as part of their program to protect personnel from ionizing radiation. The programs have been tested against the previous method, which used a graphics system on the laboratory's main computer. These programs produce equivalent results.

Approved by:

Ira M Blatstein

IRA M. BLATSTEIN, Head
Radiation Division

CONTENTS

ILLUSTRATIONS

TABLES

INTRODUCTION

This document explains in detail a group of utility and special purpose computer programs. The programs were written to be used with a microcomputer, with a serial port, and to drive a digital plotter. There are programs in this document that can be used to plot paired laboratory data and functions. There are also programs to draw ellipses, regular polygons, circles, and pie graphs. There are two polar plotting programs. One polar program to plot data and another to plot a polar function.

The programs were written specifically to drive a Houston Hiplot digital plotter with a serial interface. The plotter is a model DPM-43 designed to hold standard typing paper. The plotter is also capable of plotting in six different colors. This feature adds a nice touch to line drawings for reports. Changing the type of pen and using Mylar film instead of paper allows the plotter to produce a transparency for overhead projection.

The programs in this document use plotting software commands specific to the Hiplot plotter and were written in Northstar Basic. The program can be easily rewritten to conform to any particular form of the Basic language. The principles used in the plotting algorithms can be applied universally in any computer to plotter relationship.

This document will describe, discuss, and list these programs and provide examples of the drawings produced. These programs are interactive and user friendly so a great deal of programming expertise is not necessary in order to use them.

COMMUNICATING WITH THE HIPLOT PLOTTER

This section deals with certain elements of the programs in this document. These elements are necessary for proper communication with the plotter. These programs were written to control the operation of a Houston Instrument Hiplot plotter. This plotter uses an asynchronous, bit serial, RS-232-C level I/O port that allows the plotter to be utilized in an on-line, direct connect manner to a host computer via a plotter interface.

This section will not be a complete discussion of all the software commands necessary to drive the Hiplot. If interested in-depth information is available in the Hiplot Digital Plotter Operators Instruction Manual.

This section will discuss program elements that are the same in every program to be discussed. These elements are absolutely necessary to write anything but the most very trivial program to operate the plotter.

The most important element is a small routine that handles handshaking. The handshake routine allows a mutual cooperation between the plotter and the host computer. This is needed because the plotter has a small amount of memory storage in comparison with the host computer. The plotter can only receive so many instructions before its memory is full. The plotter is also a mechanical device so the rate at which the instructions are accomplished is slow compared with the rate the host computer can send more instructions. The plotter will respond with an ascii character 13 when ready to receive more program instructions. If the handshake routine detects any other ascii character, the routine effectively tells the host computer that the plotter memory is full and is unable to receive more instructions. If the user plans to write new programs for the Hiplot, he can use the handshake routine shown. He must be aware that the strings X$ and Q$, must be dimensioned using an appropriate dimension statement.

There are two strings that are used in every program in this document. The first of these strings is A$ = " ;: IOD 0 A ". The first part of the string, the semicolon and the colon mean select the plotter. The IOD section means use ascii character 13 as a ready for more instructions signal for handshaking. The following 0 and A mean use no time delay and use absolute coordinates. In the string the 0's are zeros not "oh's."

The second string is E$ = CHR$(95) and its purpose is to provide the plotter with an instruction that means exit the symbol drawing mode. This string is found at the end of any instruction that marks symbols, or writes words for titles.

THE HANDSHAKE ROUTINE

```
1410 DEF FNA(X$)
1420 X$=X$+CHR$(13)
1430 FOR X=1 TO LEN(X$)
1440 Y=FNO(X$(X,X))
1450 NEXT X
1460 REM
1470 Z = INT(INP(5)/2)\!Z,
1480 IF Z = 2 * INT(Z/2) THEN 1470
1490 X=INP(4)
1500 !"*",X,
1510 RETURN 0
1520 FNEND
1530 DEF FNO(Q$)
1540 Z=INP(5)
1550 IF Z/2=INT(Z/2) THEN 1540
1560 OUT 4,ASC(Q$)
1570 !Q$,
1580 IF Q$=CHR$(13) THEN !
1590 RETURN 0
1600 FNEND
```

DATPLOT

Datplot is a program designed to plot paired laboratory data. This program has many features that add reliability and distinction to its output.

As previously discussed, the program is written for a color plotter. As such Datplot will ask the user to input a choice of color for the main title, the title of the abscissa and ordinate, the abscissa scale, ordinate scale, data mark, and data plot. There is also a provision to easily input the main title and labels for both axes. The format for the input of color is a P followed by an integer 1 through 6.

Datplot has a convenient feature to relieve the user of entering his data each and every time a graph is required. This feature is the ability to save data in a known place called a disk file. Saving data on a disk allows the user to accumulate a library of data. This library not only serves as a repository for experimental data but also allows the user to recall the data at will and duplicate the drawing. The reader can refer to Appendix A for a discussion of disk files.

After the data is entered, the user can sort it. The sort can be done either on the x value carrying along the associated y value or on the y value carrying along the associated x value. A question could be raised as to the need for a sort routine at all. Why not plot the data as entered? In response, examine the following situation. The objective is to plot experimental data that has not been acquired in numeric order. For instance, an experimenter is interested in investigating the thermal expansion of a material and plotting the resulting data. Paired measurements of temperature and the corresponding length are taken. The experimental equipment is brought to a certain temperature and the length measured. Then the temperature is increased and the sample temperature and length are measured again. A better experimental technique would be to approach the desired temperature sometimes from below and sometimes from above. This process would be continued until the experimenter was satisfied that sufficient data had been accumulated.

Now if the data were to be loaded into Datplot and not sorted the user would find that the plotting routine gave an incorrect rendering of the data. To correct this, the array of independent variables x must be arranged either in ascending or descending order. That is, the data must be sorted.

Table 1 shows Datplot's ability to sort 39 pairs of data. The resulting plot of this data is shown in Figure 1. The program as listed in this publication will provide a line printer listing like Table 1. The author

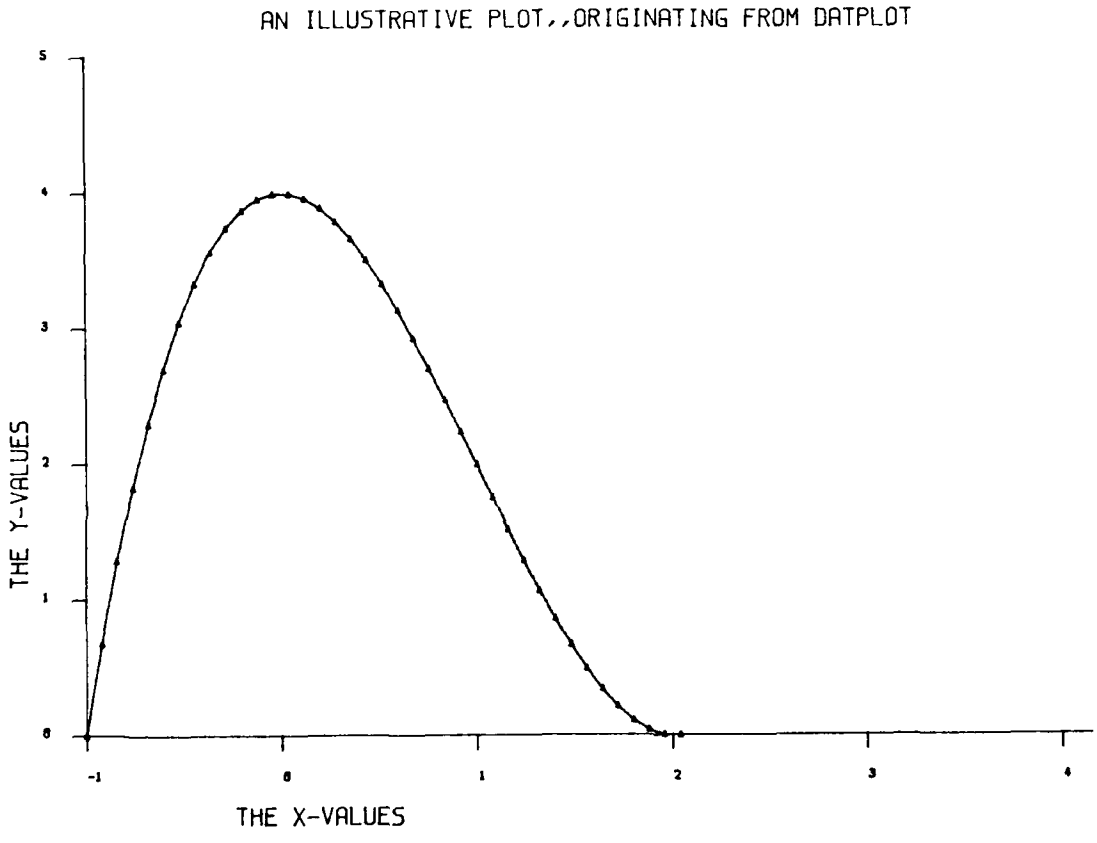


FIGURE 1. THE OUTPUT FROM DATPLOT

TABLE 1. COMPARISON OF DATA SORT TO UNSORT

UNSORTED DATA		SORTED DATA	
X VALUES	Y VALUES	X VALUES	Y VALUES
2.04	0	-1	0
1.32	1.073	-.92	.682112
1.56	.495616	-.84	1.290496
.68	2.927232	-.76	1.828224
.92	2.239488	-.68	2.298368
.12	3.958528	-.6	2.704
-1	0	-.52	3.048192
-.44	3.334016	-.44	3.334016
-.76	1.828224	-.36	3.564544
-.28	3.742848	-.28	3.742848
-.04	3.995136	-.2	3.872
.04	3.995264	-.12	3.955072
.36	3.657856	-.04	3.995136
.52	3.329408	.04	3.995264
.76	2.706176	.12	3.958528
1.08	1.760512	.2	3.888
1.88	.041472	.28	3.786752
1.72	.213248	.36	3.657856
1.48	.670592	.44	3.504384
-.68	2.298368	.52	3.329408
-.12	3.955072	.6	3.136
.28	3.786752	.68	2.927232
.6	3.136	.76	2.706176
1	2	.84	2.475904
1.24	1.293824	.92	2.239488
4	.864	1	2
	.342144	1.08	1.760512
	.112	1.16	1.524096
	.004736	1.24	1.293824
.2	3.888	1.32	1.073
-.2	3.872	1.4	.864
-.92	.682112	1.48	.670592
-.6	2.704	1.56	.495616
-.84	1.290496	1.64	.342144
-.36	3.564544	1.72	.213248
-.52	3.048192	1.8	.112
.44	3.504384	1.88	.041472
.84	2.475904	1.96	.004736
1.16	1.524096	2.04	0

ELLIPSE

The digital plotter user might be interested in drawing an isometric or perspective view of an object. The plotting routine called ellipse was written to aid the user of the program in drawing figures of the type described above. Ellipse does not draw a complete figure of an object. Its singular purpose like its title is to draw an ellipse.

The use of this program has control over four parameters, the size of the ellipse, where on the paper it is drawn, the color, and how close an approximation to a mathematical function the resulting drawing really is.

The size of the ellipse to be drawn is determined by the operator entering a value for an x intercept and a y intercept. These values should be entered in inches with fractions converted to a decimal equivalent. Refer to Figure 6 for an aid in understanding how this is used in dimensioning an ellipse in this program. Program lines 50 and 60 refer to the input of size information.

Where the ellipse is drawn on the page requires the user to enter a positional coordinate pair that will describe the geometric center of the ellipse. Program lines 70 and 80 refer to this information. Figure 7 will provide the user with a guide to how a page is arranged in terms of plotter coordinates.

Pen color is entered in the prediscussed manner, a letter P followed by an integer ranging from 1 through 6.

The program listing prompt in line 100 asks the user to choose an increment for plotting the ellipse. The term increment affects the plot in the following manner. A mathematic equation is used to produce points that are processed and converted to a pen position in the drawing. The more points that are produced the closer the drawing will approximate the equation. The finer the increment the nicer the ellipse will appear. The user can use a simple rule of thumb to insure a quality drawing. The user could enter an increment as fine as the number of decimal places used for the size values. For example a user entered a value for an x intercept of 1.063 inches and a value for the y intercept of 2.550 inches. An increment that would insure a very good drawing of an ellipse would be a value of .001.

```
1310 DEF FNA(X$)
1320 X$=X$+CHR$(13)
1330 FOR X=1 TO LEN(X$)
1340 Y=FNO(X$(X,X))
1350 NEXT X
1360 REM
1370 Z = INT(INP(5)/2)\!Z,
1380 IF Z = 2 * INT(Z/2) THEN 1370
1390 X=INP(4)
1400 !"*",X,
1410 RETURN 0
1420 FNEND
1430 DEF FNO(Q$)
1440 Z=INP(5)
1450 IF Z/2=INT(Z/2) THEN 1440
1460 OUT 4,ASC(Q$)
1470 !Q$,
1480 IF Q$=CHR$(13) THEN !
1490 RETURN 0
1500 FNEND
```

```
660 Z=FNA(A$+N$+" "+"D"+" "+STR$(J)+",150"+" "+STR$(J)+",125")
670 Z=FNA(A$+N$+" "+"U"+" "+STR$(J)+",75"+" "+"U")
680 Z=FNA(A$+N$+" "+"S11"+STR$(X1+B)+" "+E$)
690 Q=B+C
700 B=Q
710 NEXT J
720 FOR J = 150 TO 1300 STEP F
730 Z=FNA(A$+J$+" "+"U"+"150,"+STR$(J)+" "+"U")
740 Z=FNA(A$+J$+" "+"D"+" "+"150,"+STR$(J)+ " "+"125,"+STR$(J))
750 Z=FNA(A$+J$+" "+"U"+" "+"75,"+STR$(J)+" "+"U")
760 Z=FNA(A$+J$+" "+"S11"+STR$(Y1+A)+" "+E$)
770 P2=A+E
780 A=P2
790 NEXT J
800 INPUT " DO YOU WISH TO MARK DATA OR PLOT(MARK,PLOT)",R$
810 IF R$="MARK" THEN 820 \ IF R$="PLOT" THEN 950
820 REM SECTION TO MARK ORIGINAL DATA-GENERATED BY FUNCTION
830 FOR R = 0 TO N
840 V=1650/(X2-X1)
850 L = 1150/(Y2-Y1)
860 H1=P(R,1)
870 H2=V*(H1-X1)
880 H3=H2+150
890 H2=INT(H3)
900 H5=P(R,2)   \H6=L*(H5-Y1)\H7=H6+150
910 H6=INT(H7)
920 Z=FNA(A$+U$+" "+"U"+STR$(H2)+STR$(H6))
930 Z=FNA(A$+U$+" "+"M14"+" "+E$)
940 NEXT R\GOTO 800
950 REM SECTION FOR ITERPOLATION AND PLOTTING
960 FOR I = 0 TO N-1
970 X3(I)=P(I,1)
980 Y3(I)=P(I,2)
990 NEXT I
1000 R2=1.0/(C2+1)\K=0
1010 FOR I = 1 TO N-1\K=K+1
1020 X2(K)=X3(I-1)\Y2(K)=Y3(I-1)
1030 X4=(X3(I)-X3(I-1))*R2\Y4=(Y3(I)-Y3(I-1))*R2
1040 J2=1
1050 J2=J2+1
1060 K=K+1
1070 Y2(K)=Y2(K-1)+Y4
1080 X2(K)=X2(K-1)+X4
1090 IF J2<=C2 THEN 1050
1100 NEXT I\K=K+1
1110 X2(K)=X3(N-1)
1120 Y2(K)=Y3(N-1)
1130 V=1650/(X2-X1)\L=1150/(Y2-Y1)
1140 H2=V*(X2(1)-X1)\H3=H2+150
1150 H2=INT(H3)\H5=L*(Y2(1)-Y1)
1160 H6=H5+150\H5=INT(H6)
1170 Z=FNA(A$+K$+" "+"U"+" "+STR$(H2)+STR$(H5))
1180 FOR I = 1 TO K
1190 X4=X2(I)
1200 Y4=    Y2(I)
1210 V=1650/(X2-X1)
1220 L=1150/(Y2-Y1)
1230 H2=V*(X4-X1)
1240 H3=H2+150\H2=INT(H3)
1250 H5=L*(Y4-Y1)
1260 H6=H5+150\H5=INT(H6)
1270 Z=FNA(A$+K$+" "+"D"+" "+STR$(H2)+STR$(H5))
1280 NEXT I
1290 Z=FNA(A$+"PO")
1300 END
```

THE PROGRAM LISTING FOR FUNCPLOT

```
10 REM FUNCTION PLOTTER
20 REM WRITTEN BY MARSHAL PHIN JULY 83
30 E$=CHR$(95)
40 INPUT " ENTER MIN. X-VALUE FOR FUNCTION EVALUATION ",Z6\!
50 INPUT " ENTER MAX X-VALUE FOR FUNCTION EVALUATION ",Z7\!
60 INPUT " ENTER THE INCREMENT FOR X-VALUES ",P1\!
70 N=(Z7-Z6)/P1
80 IF N<=190 THEN 110
90 IF N> 190 THEN N=190
100 P1=(Z7-Z6)/N
110 C2=1
120 IF C2=N THEN Z2=C2*N\IF N>C2 THEN Z2=((N*C2)+(N-C2))
130 IF C2>N THEN Z2=((N*C2)-(C2-N))
140 DIM J$(2),N$(2),U$(2),K$(2),R$(4),Z$(3)
150 DIM P(N,2)
160 DIM X2(Z2),Y2(Z2),X3(N),Y3(N)
170 DIM A$(80),X$(80),O$(80),Q$(80),M$(80),T$(80),C$(2),D$(2),P$(2),S$(2)
180 A$=";: IOD 0 A "
190 INPUT " ENTER MAIN TITLE ",M$\PRINT
200 INPUT " PICK PEN COLOUR FOR MAIN TITLE ",C$\PRINT
210 INPUT " ENTER ABSCISSA TITLE ",O$\PRINT
220 INPUT " PICK A PEN COLOUR FOR ABSCISSA TITLE ",D$\PRINT
230 INPUT " ENTER ORDINATE TITLE ",T$\PRINT
240 INPUT " PICK PEN COLOUR FOR ORDINATE TITLE ",P$\PRINT
250 INPUT " PICK PEN COLOUR FOR AXIS ",S$\PRINT
260 INPUT " PICK A PEN COLOUR FOR ORDINATE SCALE ",J$\PRINT
270 INPUT " PICK A PEN COLOUR FOR ABSCISSA SCALE ",N$\PRINT
280 INPUT " PICK A PEN COLOUR FOR DATA MARK ",U$\PRINT
290 INPUT " PICK A PEN COLOUR FOR DATA PLOT ",K$\PRINT
300 INPUT " ENTER SCALE FACTOR FOR X ",C\PRINT
310 INPUT " ENTER SCALE FACTOR FOR Y ",E\PRINT
320 J=0
330 FOR X = Z6 TO Z7 STEP P1
340 Y=(((X*X)*X)-3*(X*X))+4
350 P(J,1)=X\P(J,2)=Y
360 J=J+1
370 NEXT X
380 INPUT " FOR GRAPH SCALE ENTER MIN X VALUE ",X1\PRINT
390 INPUT " FOR GRAPH SCALE ENTER MAX X VALUE ",X2\PRINT
400 INPUT " FOR GRAPH SCALE ENTER MIN Y VALUE ",Y1\PRINT
410 INPUT " FOR GRAPH SCALE ENTER MAX Y VALUE ",Y2\PRINT
420 REM START TO PLOT MAIN TITLE AND ABSCISSA AND ORDINATE TITLES
430 Z=FNA(A$+C$+" "+"U"+" "+"450,1350"+" "+"U")
440 Z=FNA(A$+C$+" "+"S12"+" "+M$+E$)
450 Z=FNA(A$+D$+" "+"U"+" "+"400,0"+" "+"U")
460 Z=FNA(A$+D$+" "+"S12"+" "+O$+E$)
470 Z=FNA(A$+P$+" "+"U"+" "+"50,400"+" "+"U")
480 Z=FNA(A$+P$+" "+"S42"+" "+T$+E$)
490 REM END TITLES
500 REM PLOT AXIS
510 Z=FNA(A$+S$+ " "+"U"+" "+"150,150"+" "+"U")
520 Z=FNA(A$+S$+" "+"D"+" "+"150,150"+" "+"150,1300")
530 Z=FNA(A$+S$+" "+"U"+" "+"150,150"+" "+"U")
540 Z=FNA(A$+S$+" "+"D"+" "+"150,150"+" "+"1850,150"+" "+"U")
550 REM SECTION THAT DRAWS TICK MARKS AND LABELS AXIS
560 G=((X2-X1)/C)
570 H=((Y2-Y1)/E)
580 T=(1300-150)/H
590 F=INT(T)
600 P=(1800-150)/G
610 A=0
620 B=0
630 D=INT(P)
640 FOR J = 150 TO 1800 STEP D
650 Z=FNA(A$+N$+' "+"U"+" "+STR$(J)+",150"+" "+"U")
```

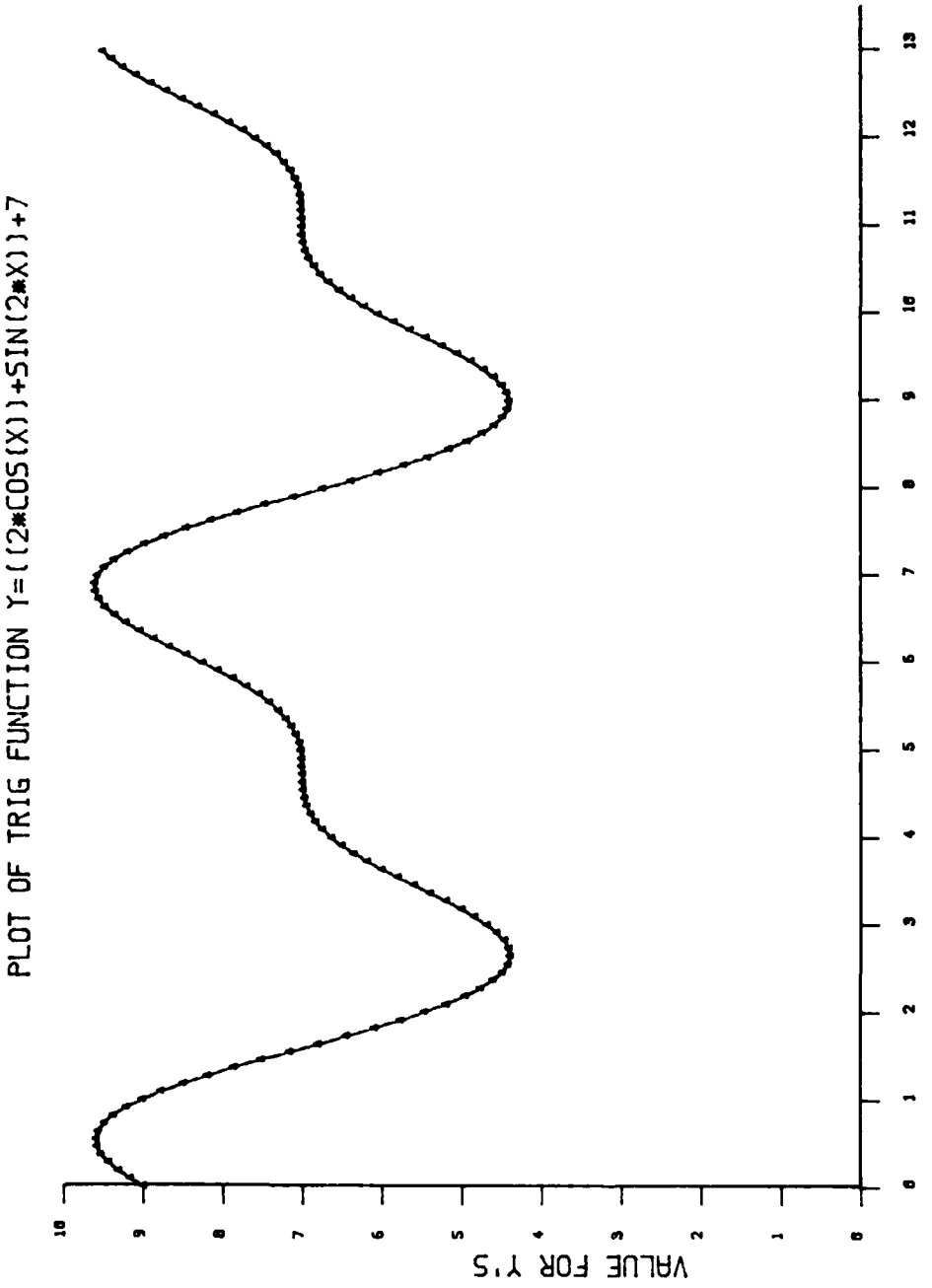


FIGURE 5. OUTPUT FROM FUNCPLOT—DIFFERENT FUNCTION MARKED AND PLOTTED

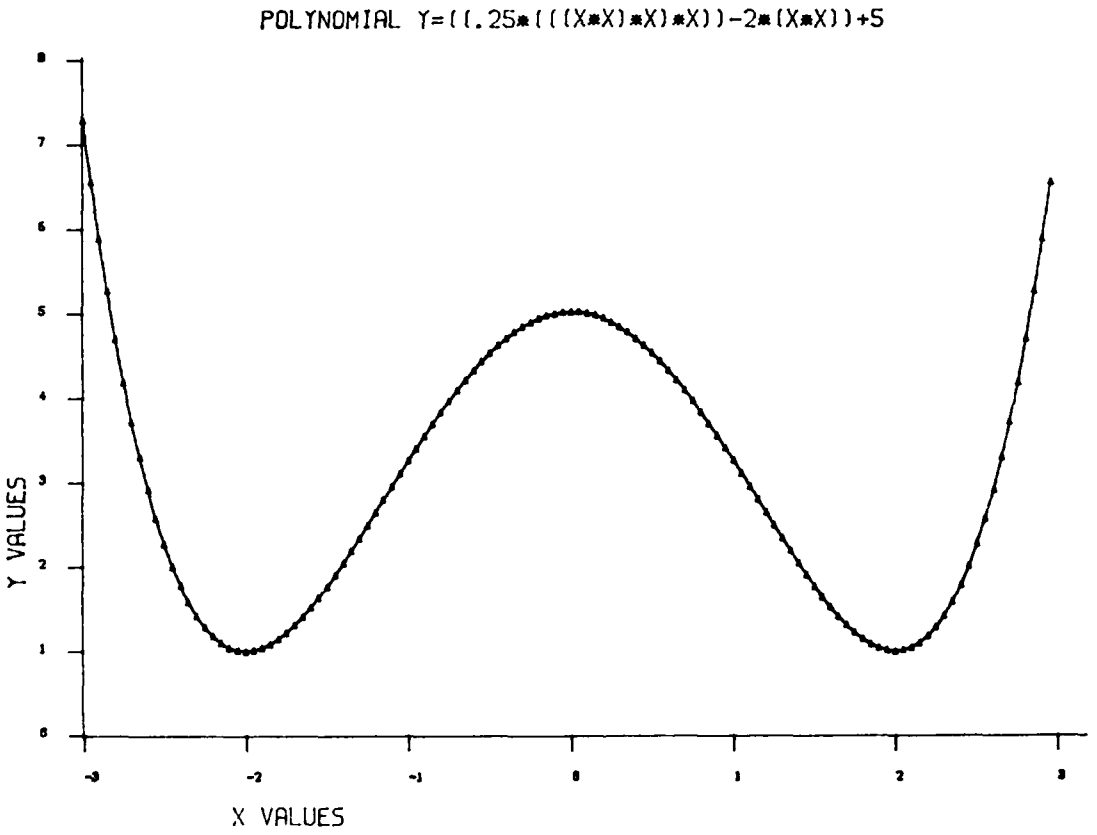


FIGURE 4. THE OUTPUT FROM FUNCPLOT--DIFFERENT FUNCTION MARKED AND PLOTTED

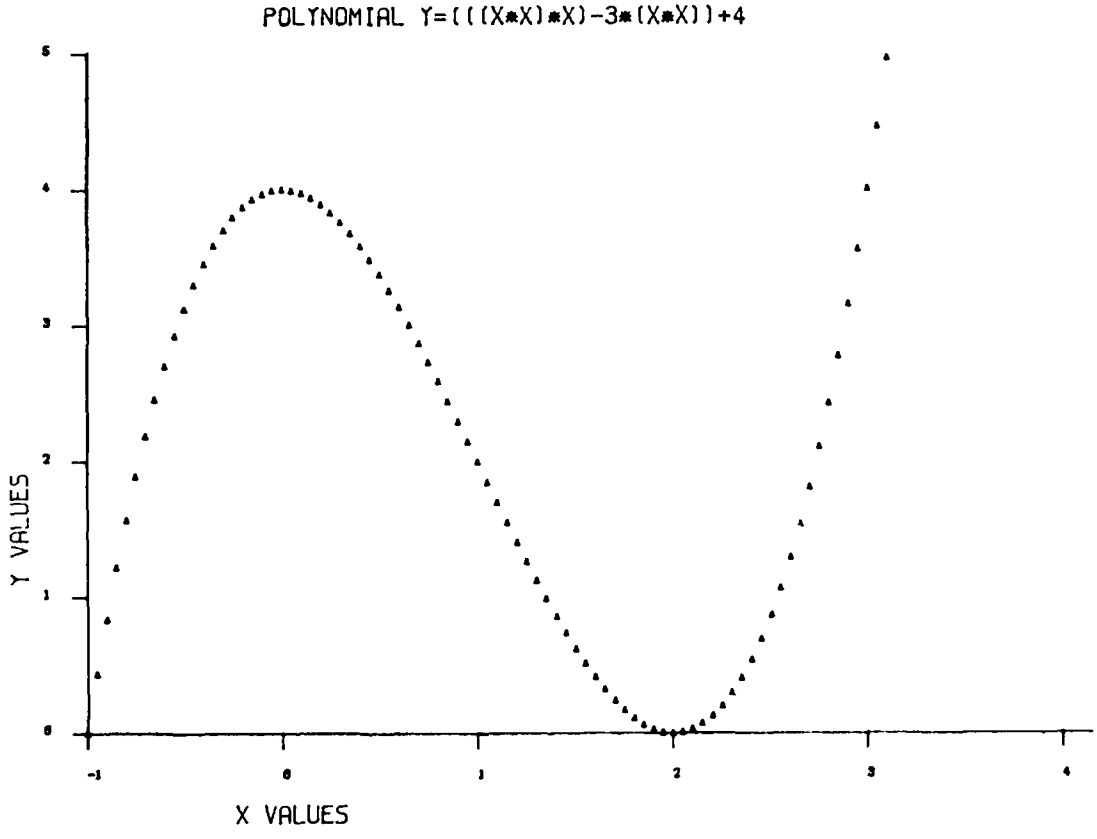


FIGURE 3. THE OUTPUT FROM FUNCPLOT--MARKED ONLY

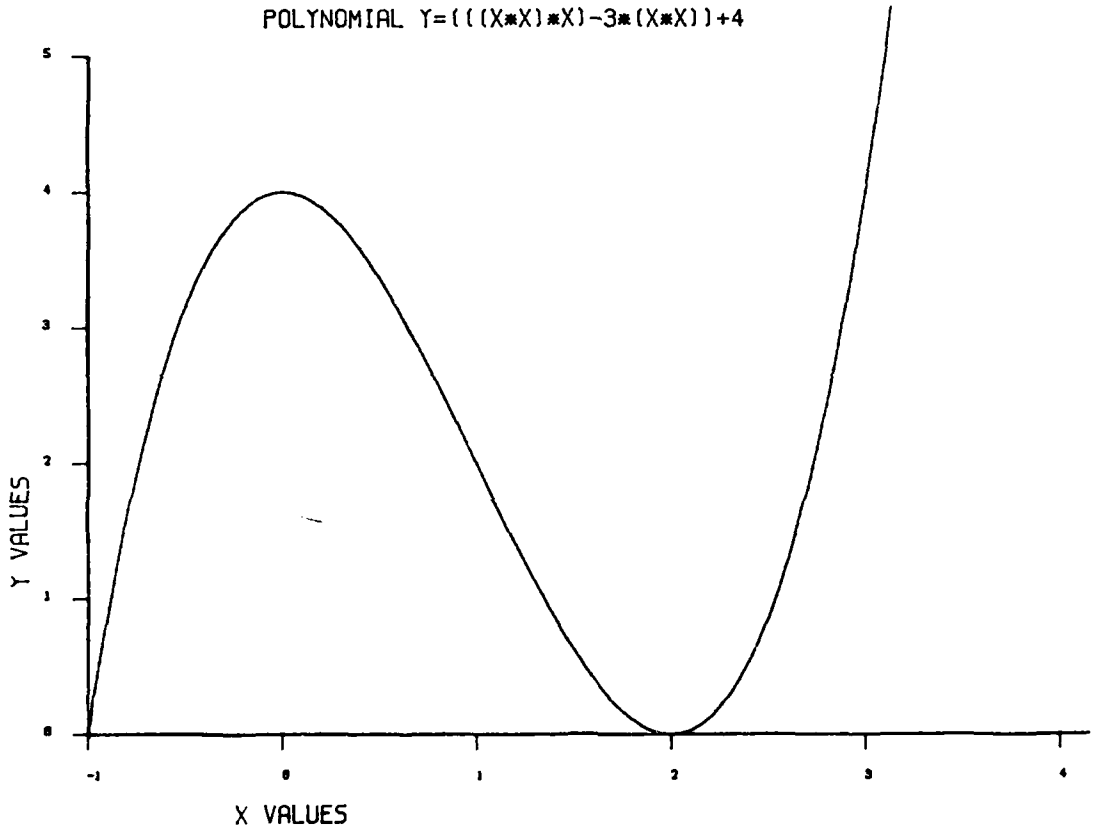


FIGURE 2. THE OUTPUT FROM FUNCPLOT-- PLOTTED ONLY

FUNCPLOT

Funcplot is a program very similar to Datplot in all but a very elementary characteristic. While Datplot asks the user to input a collection of data pairs that could approximate a function to be plotted, Funcplot in effect does the inverse. This program uses a function chosen by the user to generate data pairs that are processed and plotted to form a line drawing.

To plot different functions one line must be edited in the program. The line that defines the function is line 340. This is the line that must be changed for different functions to be plotted. This line will not look like a standard math equation; however, this is a standard manner for arithmetic equations to be expressed for computer evaluation. This is the only awkwardness to this program. Users must know how to edit at least one line of text.

The user will be asked to enter the minimum and maximum x value for the evaluation of the function. The lines that accomplish this are 40 and 50 in the program listing. The minimum and maximum values for x will describe the range of values for the independent variable. In addition the entry of an increment value is made. The line associated with this request is 60 of the program listing. The increment value will actually define how many different values for the dependent variable are produced. The finer the increment the closer the approximation to the function will be and the nicer the drawing will look.

Figures 2 through 5 are examples of the quality and format of the output obtained from Funcplot.

```
1990 H2=V*(X4-X1)
2000 H3=H2+150\H2=INT(H3)
2010 H5=L*(Y4-Y1)
2020 H6=H5+150\H5=INT(H6)
2030 Z=FNA(A$+K$+" "+"D"+" "+STR$(H2)+STR$(H5))
2040 NEXT I
2050 Z=FNA(A$+"P0")
2060 END
2070 DEF FNA(X$)
2080 X$=X$+CHR$(13)
2090 FOR X=1 TO LEN(X$)
2100 Y=FNO(X$(X,X))
2110 NEXT X
2120 REM
2130 Z = INT(INP(5)/2)\!Z,
2140 IF Z = 2 * INT(Z/2) THEN 2130
2150 X=INP(4)
2160 !"*",X,
2170 RETURN 0
2180 FNEND
2190 DEF FNO(Q$)
2200 Z=INP(5)
2210 IF Z/2=INT(Z/2) THEN 2200
2220 OUT 4,ASC(Q$)
2230 !Q$,
2240 IF Q$=CHR$(13) THEN !
2250 RETURN 0
2260 FNEND
```

```
1330 T=(1300-150)/H
1340 F=INT(T)
1350 P=(1800-150)/G
1360 A=0
1370 B=0
1380 D=INT(P)
1390 FOR J = 150 TO 1800 STEP D
1400 Z=FNA(A$+N$+" "+"U"+" "+STR$(J)+",150"+" "+"U")
1410 Z=FNA(A$+N$+" "+"D"+" "+STR$(J)+",150"+" "+STR$(J)+",125")
1420 Z=FNA(A$+N$+" "+"U"+" "+STR$(J)+",75"+" "+"U")
1430 Z=FNA(A$+N$+" "+"S11"+STR$(X1+B)+" "+E$)
1440 Q=B+C
1450 B=Q
1460 NEXT J
1470 FOR J = 150 TO 1300 STEP F
1480 Z=FNA(A$+J$+" "+"U"+"150,"+STR$(J)+" "+"U")
1490 Z=FNA(A$+J$+" "+"D"+" "+"150,"+STR$(J)+ " "+"125,"+STR$(J))
1500 Z=FNA(A$+J$+" "+"U"+" "+"75,"+STR$(J)+" "+"U")
1510 Z=FNA(A$+J$+" "+"S11"+STR$(Y1+A)+" "+E$)
1520 P2=A+E
1530 A=P2
1540 NEXT J
1550 INPUT " DO YOU WISH TO MARK DATA OR PLOT(MARK,PLOT)",R$
1560 IF R$="MARK" THEN 1570 \ IF R$="PLOT" THEN 1710
1570 REM SUB FOR MARK OF ORIGINAL DATA
1580 FOR R = 0 TO N
1590 V=1650/(X2-X1)
1600 PRINT CHR$(7)
1610 L = 1150/(Y2-Y1)
1620 H1=P(X(R),1)
1630 H2=V*(H1-X1)
1640 H3=H2+150
1650 H2=INT(H3)
1660 H5=P(X(R),2)\H6=L*(H5-Y1)\H7=H6+150
1670 H6=INT(H7)
1680 Z=FNA(A$+U$+" "+"U"+STR$(H2)+STR$(H6))
1690 Z=FNA(A$+U$+" "+"M14"+" "+E$)
1700 NEXT R\GOTO 1550
1710 REM SUB FOR INTERP AND PLOT
1720 FOR I = 0 TO N-1
1730 X3(I)=   (P(X(I),1))
1740 Y3(I)=   (P(X(I),2))
1750 NEXT I
1760 R2=1.0/(C2+1)\K=0
1770 FOR I = 1 TO N-1\K=K+1
1780 X2(K)=X3(I-1)\Y2(K)=Y3(I-1)
1790 X4=(X3(I)-X3(I-1))*R2\Y4=(Y3(I)-Y3(I-1))*R2
1800 J2=1
1810 J2=J2+1
1820 K=K+1
1830 Y2(K)=Y2(K-1)+Y4
1840 X2(K)=X2(K-1)+X4
1850 IF J2<=C2 THEN 1810
1860 NEXT I\K=K+1
1870 X2(K)=X3(N-1)
1880 Y2(K)=Y3(N-1)
1890 V=1650/(X2-X1)\L=1150/(Y2-Y1)
1900 H2=V*(X2(1)-X1)\H3=H2+150
1910 H2=INT(H3)\H5=L*(Y2(1)-Y1)
1920 H6=H5+150\H5=INT(H6)
1930 Z=FNA(A$+K$+" "+"U"+" "+STR$(H2)+STR$(H5))
1940 FOR I = 1 TO K
1950 X4=X2(I)
1960 Y4=    Y2(I)
1970 V=1650/(X2-X1)
1980 L=1150/(Y2-Y1)
```

```
670 NEXT R
680 CLOSE#1
690 !\!"NOW LETS CHECK OUR DATA STORED IN OUR DISK FILE"\!
700 OPEN#1,F$
710 J=0
720 IF TYP(1)=0 THEN 780
730 REM ABOVE CHECKS FOR END OF FILE
740 READ#1,P(J,1)\READ#1,P(J,2)
750 PRINT P(J,1),\PRINT P(J,2)
760 J=J+1
770 GOTO 720
780 !\!\INPUT "WAS DATA OK (Y OR N )",Y$
790 CLOSE #1
800 IF J<N THEN N=J
810 IF Y$="N" THEN 420
820 PRINT #2 " UNSORTED DATA "
830 PRINT#2  " ------------- "\!#2\!#2
840 PRINT#2  " X VALUES       Y VALUES"
850 PRINT#2  "----------     ----------"\!#2
860 FOR R=0 TO N-1
870 PRINT#2 P(R,1),\!#2 TAB(13),P(R,2)
880 NEXT R
890 INPUT " FOR GRAPH SCALE ENTER MIN X VALUE ",X1\PRINT
900 INPUT " FOR GRAPH SCALE ENTER MAX X VALUE ",X2\PRINT
910 INPUT " FOR GRAPH SCALE ENTER MIN Y VALUE ",Y1\PRINT
920 INPUT " FOR GRAPH SCALE ENTER MAX Y VALUE ",Y2\PRINT
930 REM START DATA SORT SECTION
940 REM
950 PRINT\ INPUT " TO SORT BY X ENTER 1,BY Y ENTER 2 ",J
960 FOR R= 0 TO N-1\ K(R)=P(R,J)\NEXT R
970 GOSUB 990
980 GOTO 1090
990 FOR A1 = 0 TO N-1
1000 P1=0
1010 FOR B1= 0 TO N-1
1020 IF K(A1)>K(B1) THEN P1=P1+1
1030 IF K(A1)=K(B1) THEN 1050
1040 GOTO 1080
1050 IF A1>B1 THEN 1070
1060 GOTO 1080
1070 P1=P1+1
1080 NEXT B1\X(P1)=A1\NEXT A1\RETURN
1090 PRINT#2 CHR$(12)
1100 PRINT#2 " SORTED DATA "
1110 PRINT#2 "-------------"\!#2\!#2
1120 PRINT#2 " X VALUES        Y VALUES"
1130 PRINT#2 "----------      ----------"\!#2
1140 FOR R=0 TO N-1
1150 PRINT#2 P(X(R),1),\!#2 TAB(13),P(X(R),2)
1160 NEXT R
1170 REM START TO PLOT MAIN TILE AND ABSCISSA AND ORDINATE TITLES
1180 Z=FNA(A$+C$+" "+"U"+" "+"450,1350"+" "+"U")
1190 Z=FNA(A$+C$+" "+"S12"+" "+M$+E$)
1200 Z=FNA(A$+D$+" "+"U"+" "+"400,0"+" "+"U")
1210 Z=FNA(A$+D$+" "+"S12"+" "+O$+E$)
1220 Z=FNA(A$+P$+" "+"U"+" "+"50,400"+" "+"U")
1230 Z=FNA(A$+P$+" "+"S42"+" "+T$+E$)
1240 REM END TITLES
1250 REM PLOT AXIS
1260 Z=FNA(A$+S$+ " "+"U"+" "+"150,150"+" "+"U")
1270 Z=FNA(A$+S$+" "+"D"+" "+"150,150"+" "+"150,1300")
1280 Z=FNA(A$+S$+" "+"U"+" "+"150,150"+" "+"U")
1290 Z=FNA(A$+S$+" "+"D"+" "+"150,150"+" "+"1850,150"+" "+"U")
1300 REM SECTION THAT DRAWS TICK MARKS AND LABELS AXIS
1310 G=((X2-X1)/C)
1320 H=((Y2-Y1)/E)
```

THE PROGRAM LISTING FOR DATPLOT

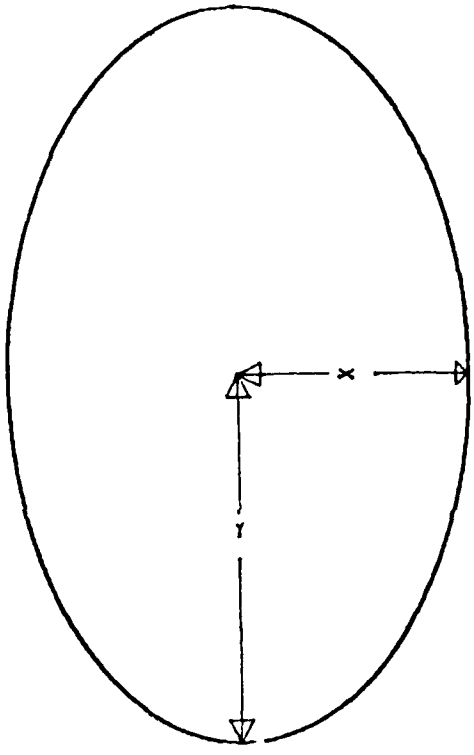
```
10 REM DATA PLOTTER
20 REM WRITTEN BY MARSHAL PHIN JULY 83
30 E$=CHR$(95)
40 DIM Y$(1),F$(4)
50 PRINT" THE NEXT QUERY REGARDS READING A FILE-VS-ENTERING FRESH DATA"
60 INPUT " ARE YOU GOING TO ENTER NEW DATA (Y OR N)",Y$
70 IF Y$="N" THEN 90
80 GOTO 100
90 N=80 \GOTO 110
100 INPUT " ENTER HOW MANY DATA   PAIRS ",N\PRINT
110   INPUT " ENTER NUMBER OF DATA POINTS BETWEEN ORIGINAL ",C2
120 IF C2=N THEN Z2=C2*N\IF N>C2 THEN Z2=((N*C2)+(N-C2))
130 IF C2>N THEN Z2=((N*C2)-(C2-N))
140 DIM J$(2),N$(2),U$(2),K$(2),R$(4),Z$(3)
150 DIM P(80,2),K(80),X(80)
160 DIM X2(Z2),Y2(Z2),X3(N),Y3(N)
170 DIM A$(80),X$(80),O$(80),Q$(80),M$(80),T$(80),C$(2),D$(2),P$(2),S$(2)
180 A$=";: IOD 0 A "
190 INPUT " ENTER MAIN TITLE ",M$\PRINT
200 INPUT " PICK PEN COLOUR FOR MAIN TITLE ",C$\PRINT
210 INPUT " ENTER ABSCISSA TITLE ",O$\PRINT
220 INPUT " PICK A PEN COLOUR FOR ABSCISSA TITLE ",D$\PRINT
230 INPUT " ENTER ORDINATE TITLE ",T$\PRINT
240 INPUT " PICK PEN COLOUR FOR ORDINATE TITLE ",P$\PRINT
250 INPUT " PICK PEN COLOUR FOR AXIS ",S$\PRINT
260 INPUT " PICK A PEN COLOUR FOR ORDINATE SCALE ",J$\PRINT
270 INPUT " PICK A PEN COLOUR FOR ABSCISSA SCALE ",N$\PRINT
280 INPUT " PICK A PEN COLOUR FOR DATA MARK ",U$\PRINT
290 INPUT " PICK A PEN COLOUR FOR DATA PLOT ",K$\PRINT
300 INPUT " ENTER SCALE FACTOR FOR X ",C\PRINT
310 INPUT " ENTER SCALE FACTOR FOR Y ",E\PRINT
320 INPUT" DO YOU WISH TO CREATE A DATA FILE ( Y OR N )",Y$\!
330 IF Y$="N" THEN 370
340 INPUT " ENTER NAME OF FILE TO CREATE ",F$\!
350 INPUT " ENTER BLOCK SIZE FOR MEMORY ALLOTTMENT ",B1\!\!
360 CREATE F$,B1
370 INPUT " DO YOU WISH TO READ A DATA LIBRARY FILE ( Y OR N )",Y$\!
380 IF Y$="N" THEN 400
390 INPUT "ENTER FILE TO BE READ",F$\GOTO 690
400 INPUT " DO YOU WISH TO USE THE DEFAULT FILE OR NAME YOUR OWN(Y OR N)",Y$
410 IF Y$="N" THEN 600
420 OPEN #1, "PLOT"
430 FOR R=1 TO N
440 INPUT"ENTER PLOT DATA FOR X ",P(R,1)
450 INPUT"ENTER PLOT DATA FOR Y ",P(R,2)
460 WRITE#1,P(R,1)
470 WRITE#1,P(R,2)
480 NEXT R
490 CLOSE#1
500 !\!"NOW LETS CHECK OUR DATA STORED ON THE DISK"\!
510 OPEN#1,"PLOT"
520 IF TYP(1)=0 THEN 570
530 REM ABOVE CHECKS FOR END OF FILE
540 READ#1,I
550 PRINT I,\PRINT I
560 GOTO 520
570 !\!\INPUT "WAS DATA OK (Y OR N )",Y$
580 CLOSE #1
590 IF Y$="N" THEN 420
600 INPUT " ENTER THE NAME OF YOUR DATA FILE ",F$
610 OPEN#1 ,F$
620 FOR R=1 TO N
630 INPUT"ENTER DATA FOR X ",P(R,1)
640 INPUT"ENTER DATA FOR Y ",P(R,2)
650 WRITE#1,P(R,1)
660 WRITE#1,P(R,2)
```

intended to show in the program listing how the table was made. To suppress this listing lines 820 through 850 and 1100 through 1130 can be deleted without affecting the rest of the program.

The axes of this program are self scaling. The user inputs his choice of maximum and minimum values for both the ordinate and abscissa axes. The program supplies the user with a prompt that asks him to enter a value the program refers to as a scale factor. This input describes the interval between the tic marks that will be drawn along the axes.

After the data has been entered, sorted, and loaded in an array, it is marked and/or plotted. The maximum and minimum values chosen obviously establish the range of the data input, or at least the range the user is interested in. The maximum and minimum values are also used by the program to scale the data before it is actually marked and/or plotted. The line numbers associated with scaling the data are 1970 through 2010. In marking the data, the array is scaled one element at a time and the resulting information is translated by the plotter to a positional coordinate on the paper. The data point is then marked with a small triangle.

After marking the distinct data points, the user has the option to leave the program by typing a control C or to plot a line connecting the points. To improve the resolution of the line drawing and provide a better approximation to the behavior of the curve between data points, an interpolation routine is provided in Datplot. In essence, this routine creates more data between marked data. The user is asked to input how many data points they would like to create between marked data. The output from Datplot in Figure 1 was plotted with three points made between markings.



X = DISTANCE FROM CENTRE TO X INTERCEPT

Y = DISTANCE FROM CENTRE TO Y INTERCEPT

FIGURE 6. THE METHOD USED TO DIMENSION THE SIZE OF AN ELLIPSE

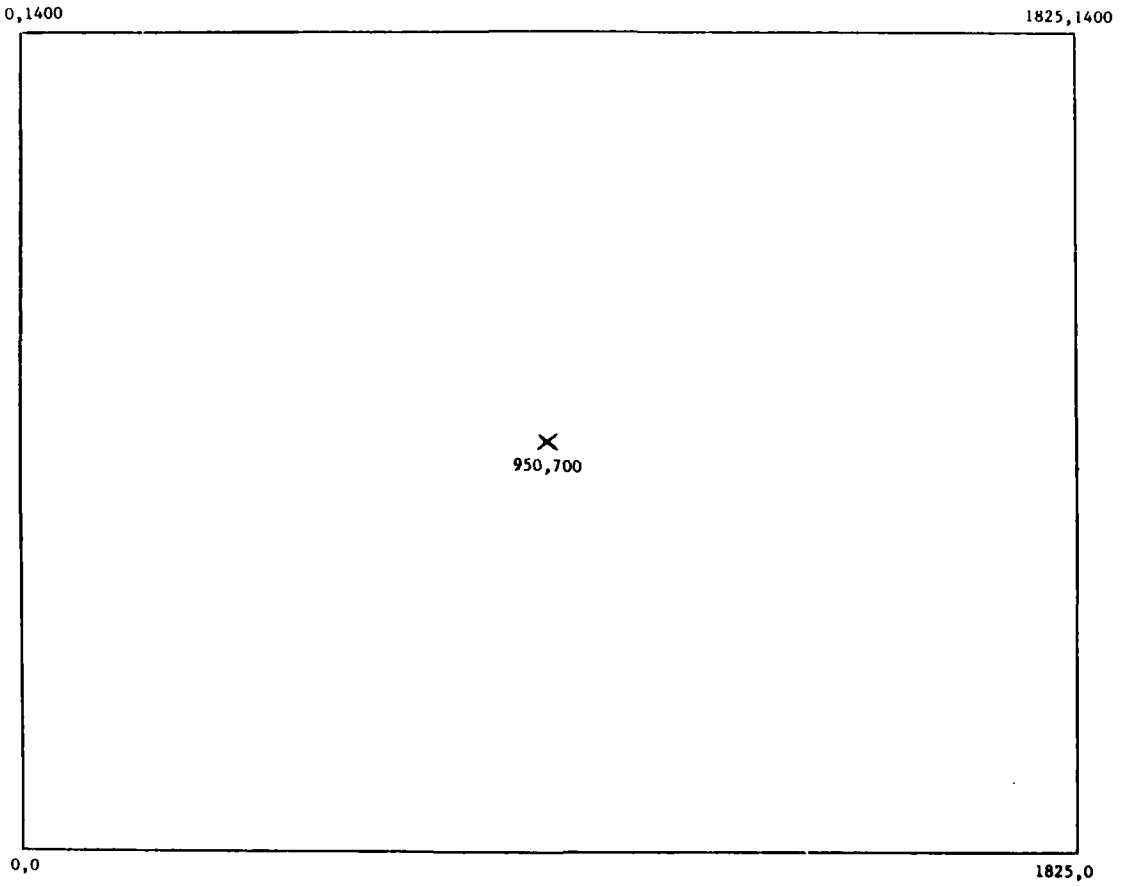


FIGURE 7. THE RANGE OF PLOTTER COORDINATES

PROGRAM LISTING FOR ELLIPSE

```
10 REM  ELLIPSE PLOTTING PROGRAM
20 REM WRITTEN BY MARSHAL PHIN JULY 83
30 DIM A$(80),X$(80),O$(80),Q$(80),C$(2),D$(4),H$(3)
40 A$=";: IOD 0 A "
50 INPUT " ENTER VALUE FOR X INTERCEPT ",A\PRINT
60 INPUT " ENTER VALUE FOR Y INTERCEPT ",B\PRINT
70 INPUT " ENTER POSITIONAL OFFSET FOR X CORD..",S\PRINT
80 INPUT " ENTER POSITIONAL OFFSET FOR Y CORD..",T\PRINT
90 INPUT " CHOOSE PEN COLOUR FOR PLOT ",C$\PRINT
100 INPUT " ENTER INCREMENT FOR PLOT ",H\PRINT
110 PRINT\PRINT\PRINT\PRINT
120 Q=(A*200)+S
125 E=INT(Q)
126 Y=(SQRT((A*A)-(A*A)))*(B/A)
127 I=(Y*200)+T
128 F=INT(I)
130 Z=FNA(A$ +"U" +C$ + STR$(E) +STR$(F))
140 FOR J= A TO -A STEP -H
150 P=(J*200)+S
155 G = INT(P)
160 Y=(SQRT((A*A)-(J*J)))*(B/A)
162 I=(Y*200)+T
163 PRINT
165 F=INT(I)
170 Z=FNA(A$+ "D" +C$+ STR$(G) +STR$(F))
180 NEXT J
181 L=(A*200)+S \L1=INT(L)
182 Y=(SQRT((A*A)-(A*A)))*(B/A)
183 I=(Y*200)+T
184 F=INT(I)
185 Z=FNA(A$+"U"+C$+STR$(L1)+STR$(F))
190 FOR J=  A TO -A STEP -H
200 P=ABS((J*200)+S)
210 G=INT(P)
220 Y=(SQRT((A*A)-(J*J)))*(B/A)
230 I=ABS((Y*200)-T)
240 F=INT(I)
250 Z=FNA(A$+"D"+C$+STR$(G)+STR$(F))
260 NEXT J
2230 DEF FNA(X$)
2240 X$=X$+CHR$(13)
2250 FOR X=1 TO LEN(X$)
2260 Y=FNO(X$(X,X))
2270 NEXT X
2280 REM
2290 Z = INT(INP(5)/2)\!Z,
2300 IF Z = 2 * INT(Z/2) THEN 2290
2310 X=INP(4)
2320 !"*",X,
2330 RETURN 0
2340 FNEND
2350 DEF FNO(Q$)
2360 Z=INP(5)
2370 IF Z/2=INT(Z/2) THEN 2360
2380 OUT 4,ASC(Q$)
2390 !Q$,
2400 IF Q$=CHR$(13) THEN !
2410 RETURN 0
2420 FNEND
```

CIRCLES

Circles is the name of a program that can be used to draw circles, geometric polygons, and pie graphs.

This program is unique from the others in this document. This uniqueness comes from the ability to provide the user with complete instructions for the use of the program. The first program prompt, line 150 in the program listing, asks the user if he would like to receive an instructional response from the program. If the user would like to read the instructions he should enter a YES, the computer will then list the instruction section on the terminal screen. The section that holds the instruction material can be found in lines 900 through 1180 of the program listing. If the user does not require instructions an entry of anything other then yes must be made.

The next program prompt will ask the user to enter the radius of the figure to be drawn. This entry should be made in inches expressed in decimal equivalence, for instance 2.375. The maximum allowed radius is 3.25 inches. There is an error trapping feature in the program that will allow the user to enter a new radius if an attempt is made to draw a figure larger then the maximum. All circles and other figures are drawn with a radius measured from the center of the page.

A following program prompt asks the user for an entry of the number of revolutions. In the program one revolution is enternally defined as 6.283 radians or twice Pi. For most applications, circles, pie graphs, and regular polygons one revolution is sufficient. There are instances, however, when the user may use many revolutions to achieve a desired affect. This would primarily be an artist or decorative application of this program. A prompt follows regarding an input described as a step increment. This input can be defined as an interval used to divide one revolution. This would define the angular displacement used by the routine that plots the points.

The ability for one program to accomplish so many visually varying drawings lies with the mathematical alogrithm used to generate the positional coordinates the pen uses to draw the figure. When the program prompts the user to enter his desired radius, the number of revolutions, and the increment by which the number of revolutions are divided, it is asking the user to define a series of points expressed in polar coordinates. These polar coordinates are converted to a series of x and y coordinate pairs. These pairs are scaled to correspond to a position on the page. The plotter moves from one to another of these positions drawing the figure.

By entering different increments the user can draw regular polygons. Figures 8 through 12 illustrate this nicely. Notice the triangle shape was drawn with an increment of 2.094 while the decagon was drawn with an increment of .628. Notice through the series that as the increment grew smaller the closer the figures approximated a circle. Therefore, to draw a very good circle the increment should be very fine. A good rule of thumb is to make the increment as fine as the number of decimal places as the radius. Figures 13 and 14 show two possible applications.

Another feature of this program is the ability to use the symbol marking capability inherent to the plotter. A program prompt, line 250 in the program listing, will ask the user if he would like to mark a symbol at the center of the figure to be drawn. These symbols are stored in RAM in the plotter. They can be used by entering the proper command when prompted by line 250. This command must follow the following format. Figure 15 describes the various symbols.

The proper format for marker plotting commands.

FORMAT:	Mhm
M	marker plotting
h	height of marker; single digit number from one to five
m	marker type; single digit number from zero to five.

Almost every person in the scientific and business community is asked however infrequently to project areas of particular expense in their periodic budget reports. Pie graphs provide a quicker, more striking way of presenting the same information that can be presented in a table.

A pie graph presents data as wedge shaped sections of a circle. The circle equals 100 percent, or the whole, of some quantity (a tax dollar, yearly expense, hours spent to accomplish a certain task), with the wedges representing the various ways in which the whole is divided. In Figure 16, for example, the circle stands for a city tax dollar, and it is divided into units equivalent to the percentage of the tax dollar spent on various services.

In the program listing, lines 490 through 880 describe the subroutine that draws pie graphs. This subroutine has many features that allow the user to easily draw a distinctive pie graph. When prompted the user should enter the desired size of the wedge, in hundredths, .1 or .05 for instance. The subroutine keeps track of how much of the circle has been used and relays this information to the user before he has to input a percentage for the next wedge.

The user can choose a different color for each sector and also shade the sector if desired to show a difference between sectors of the same color. To shade a sector the user enters an increment of shade. To illustrate this consider an example. A user would like to shade a 15 percent wedge on a pie graph. Ther user has entered .15 for the sector definition prompt and .01 for

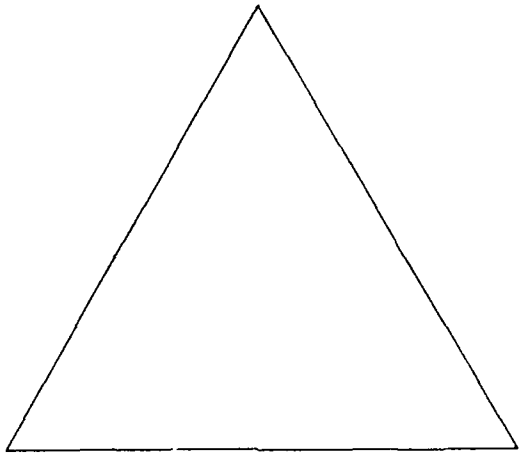


FIGURE 8. TRIANGLE ONE REVOLUTION INCREMENT OF 2.094

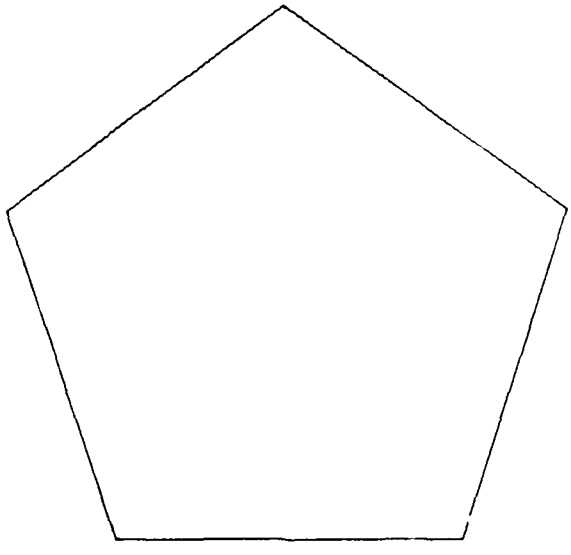


FIGURE 9. PENTAGON ONE REVOLUTION INCREMENT OF 1.256

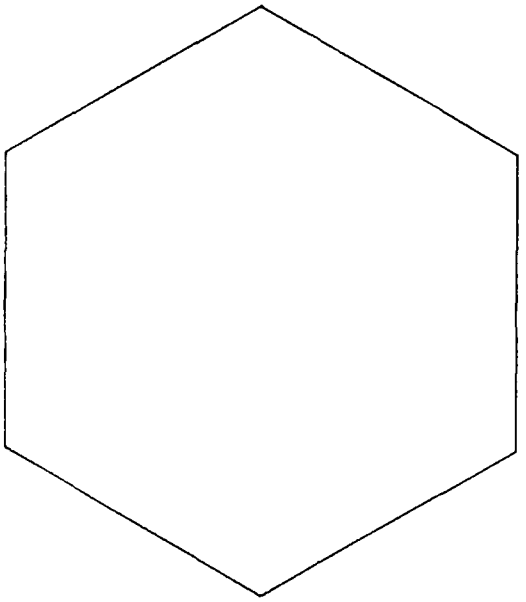


FIGURE 10. HEXAGON ONE REVOLUTION INCREMENT OF 1.047

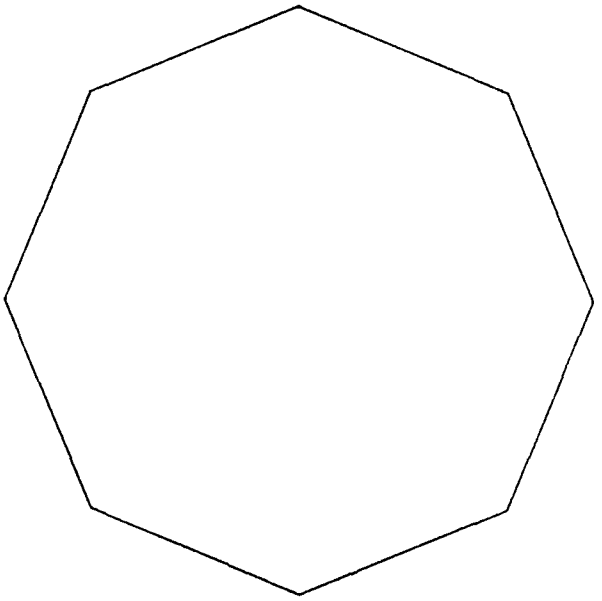


FIGURE 11. OCTAGON ONE REVOLUTION INCREMENT OF .785

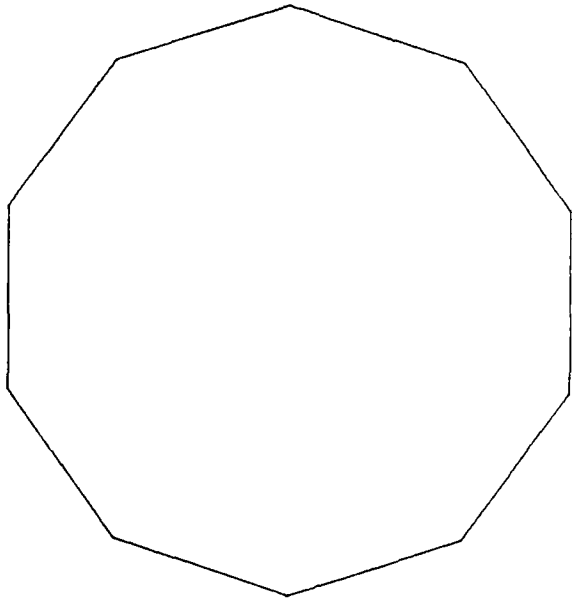


FIGURE 12. DECAGON ONE REVOLUTION INCREMENT OF .628

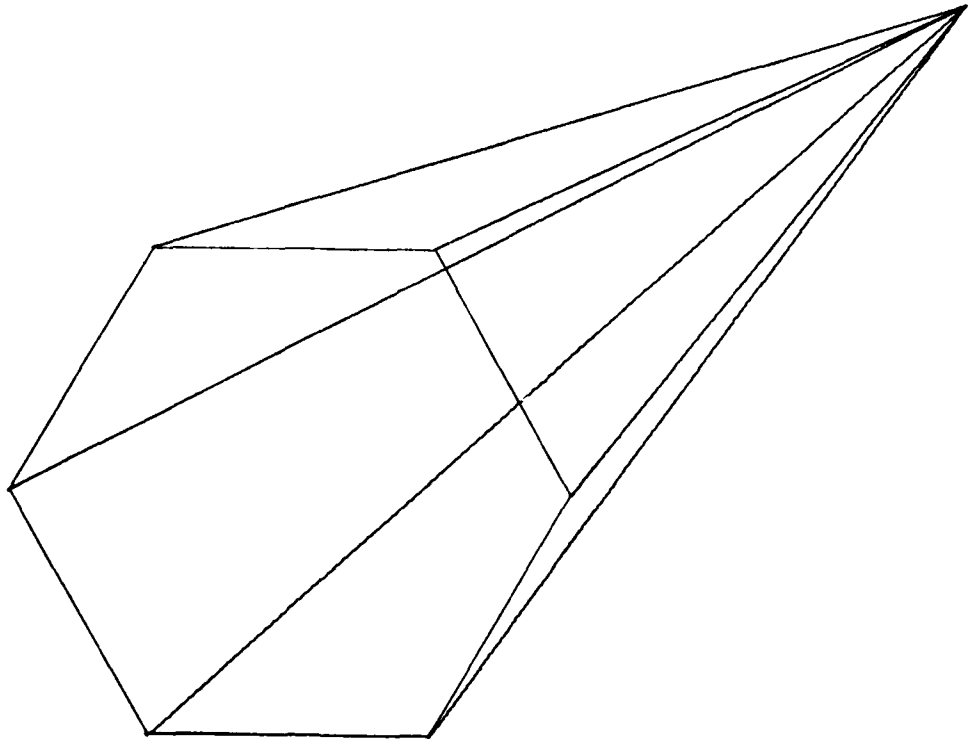


FIGURE 13. POSSIBLE APPLICATION IN PERSPECTIVE DRAWING

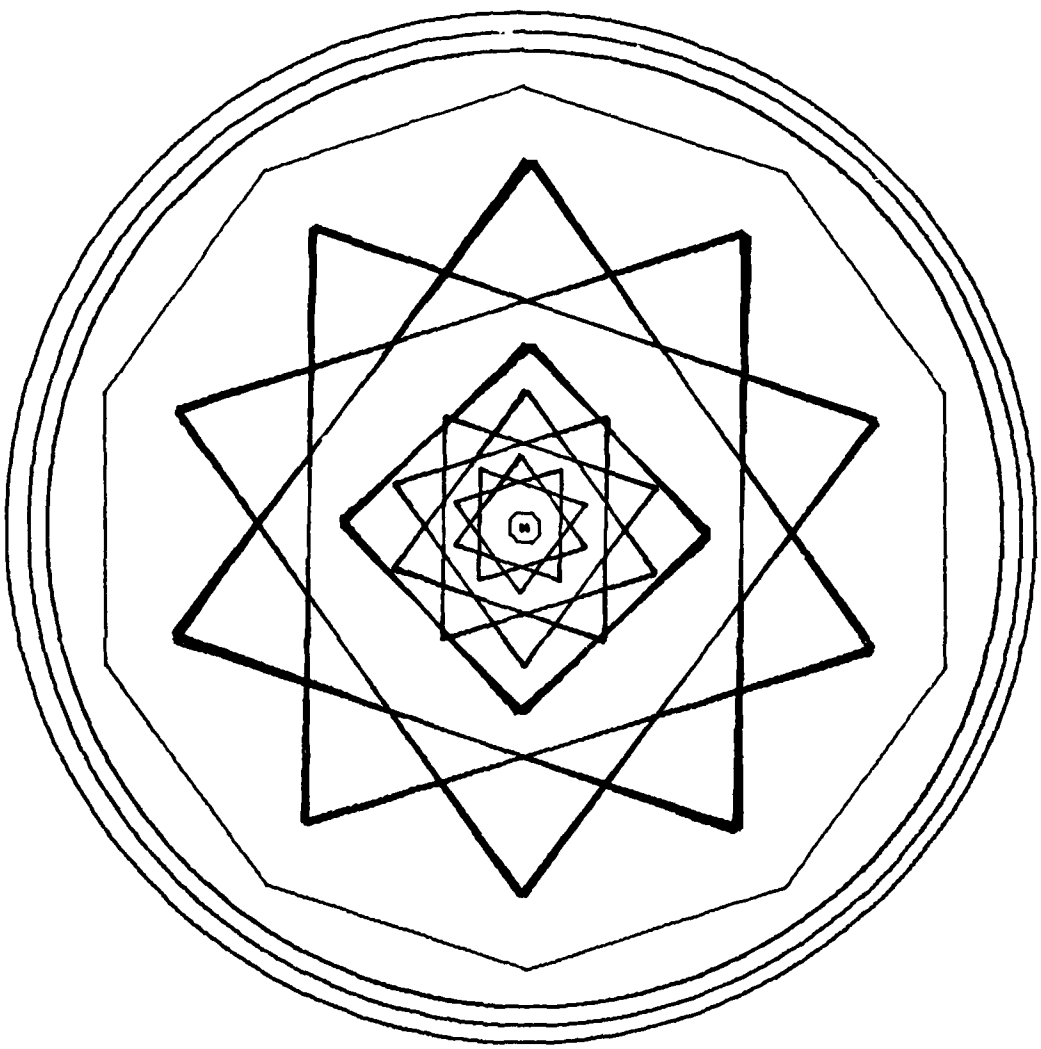


FIGURE 14. EXAMPLE OF DECORATIVE APPLICATION

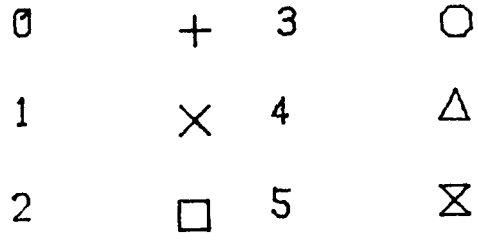


FIGURE 15. THE MARKER TYPES

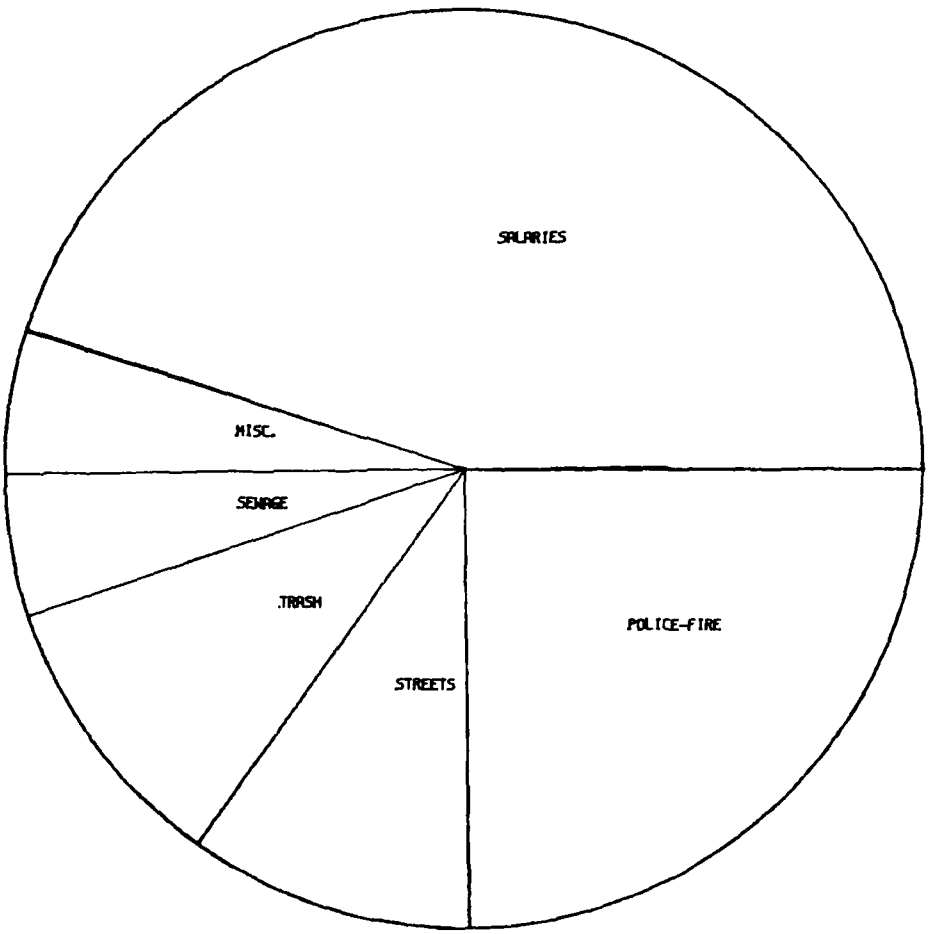


FIGURE 16. PIE GRAPH EXAMPLE---YOUR MUNICIPAL TAX DOLLAR

the shade increment. In the drawing a line will be drawn radiating from the center as long as the entered radius. There will be 15 lines drawn in this fashion to shade this wedge, one every 1 percent of the circle up to 15 percent. In this manner sector shading can be varied making sectors of the same color appear of a lighter or darker hue through the use of a coarse or fine increment.

When a user constructs a pie graph, he should keep the following things in mind. To make the percentages as clear as possible, he should sequence the wedges clockwise, from largest to smallest. Shading of the wedges should be done in a clockwise manner and from light to dark.

Although pie graphs have strong visual impact, they also have drawbacks. If more than five or six items are presented, the graph looks cluttered. Also, since they usually present percentages of something they must often be accompanied by a table listing precise statistics.

'HE PROGRAM LISTING FOR CIRCLES

```
10 REM PROGRAM TO DRAW CIRCLES ,POLYGONS,AND PIE CHARTS
20 PRINTTAB(13)," CCCC    IIIII   RRRR   CCCC   LLL   EEEEE    SSSS "
30 PRINTTAB(13),"C    C    III    R   R C    C  LLL   EE      SS  "
40 PRINTTAB(13),"CC        III    R  R  C       LLL   EEEEE     SS "
50 PRINTTAB(13),"C    C    III    R  RR C    C  LLL   EE         SS"
60 PRINTTAB(13)," CCCC    IIIII   R   R  CCCC   LLLLL EEEEE    SSSS "
70 PRINT\PRINT\PRINT\PRINT\PRINT TAB(8)," WRITTEN BY MARSHAL PHIN JUNE 83 "
80 PRINT TAB(8)," ALL RIGHT'S RESERVED "
90 PRINT\PRINT\PRINT\PRINT\PRINT\PRINT
100 W$=CHR$(43)
110 E$=CHR$(95)
120 REM
130 DIM A$(80),X$(80),O$(80),Q$(80),C$(2),D$(4),H$(3),F$(3),P$(3),I$(3),M$(1)
140 A$=";: IOD 0 A"
150 INPUT " DO YOU WANT DETAILED INSTRUCTIONS (YES OR NO)",H$
160 IF H$="YES" THEN GOSUB 890
170 INPUT " CHOOSE RADIUS IN INCHES EG. (1.0 , 2.35 ETC )",R1
180 R=R1*200
190 IF R > 650 THEN PRINT " RADIUS TO BIG "
200 IF R > 650 THEN 170
210 PRINT " ONE REV=2 PI OR 6.28 STEP .01 6.3 STEP .1 "
220 INPUT " ENTER DESIRED NUMBER OF REVOLUTION'S",J
225 J=(J*6.283)
230 INPUT " CHOOSE PEN COLOUR !EG.(P4)",C$
240 INPUT " CHOOSE STEP INCREMENT EG.(.1 OR .01 OR .75 ETC)",H
250 INPUT " DO YOU WISH TO DRAW A SYMBOL AT CENTER (YES OR NO)",P$
260 IF P$="YES" THEN GOSUB 430
270 INPUT " DO YOU WISH TO DRAW A PIE CHART (YES OR NO)",D$
280 PRINT " WARNING INCREMENT MUST BE .1 OR .01 "
290 IF D$="YES" THEN GOSUB 490
300 S=950
310 T=700
320 Q=S+R
330 P=T+R
340 Z=FNA(A$+"U"+C$+STR$(Q)+STR$(T))
350 FOR I=0 TO J STEP H
360 X=(R*COS(I))+S
370 Y=(R*SIN(I))+T
380 W=INT(X)
390 A=INT(Y)
400 Z=FNA(A$ + C$ + "D" + STR$(W)+STR$(A))
410 NEXT I
415 Z=FNA(A$+"PO")
420 END
430 INPUT " CHOOSE SYMBOL TO BE DRAWN AT CENTER ",F$
440 S=950
450 T=700
460 Z=FNA(A$+"U"+C$+STR$(S)+STR$(T))
470 Z=FNA(A$+F$+STR$(S)+STR$(T)+E$)
480 RETURN
490 REM  SUB FOR PIE CHART OPTION
500 N=0
510 Z=0
520 F=0
530 PRINT " ENTER PERCENT EXPRESSED IN HUNDRETH'S (EG. .10 FOR TEN%)"\PRINT
540 V=(E)*100
550 PRINT " PERCENT USED ",V
560 A=100-V
570 PRINT " YOU HAVE THIS MUCH PERCENT REMAINING ",A\PRINT
580 INPUT " DEFINE SECTOR IN HUNDRETH'S OF TOTAL EG.(.10)",G\PRINT
590 E=F+G
600 INPUT " CHOOSE PEN COLOR FOR SECTOR ",C$\PRINT
610 INPUT " DESIGNATE INCREMENT OF SHADE ",C\PRINT
620 INPUT " ENTER TITLE FOR SECTOR ",O$\PRINT
630 INPUT " ENTER SIZE FOR TITLES (Eg.S11)",P$\PRINT
640 INPUT " CHOOSE COLOUR FOR TITLE ",I$\PRINT
650 K=(G*6.3)+N
660 S=950
```

```
670 T=700
680 IF K >  6.3 THEN PRINT" WARNING WILL OVERWRITE FIRST SECTOR "
690 IF K >  6.3 THEN PRINT" THEREFORE I WILL ABORT ,BYE"
700 IF K >  6.3 THEN GOTO 880
710 FOR I = N TO K STEP C
720 X= (R*COS(I))+S
730 Y= (R*SIN(I))+T
740 W=INT(X)
750 A=INT(Y)
760 Z=FNA(A$+"U"+C$+STR$(S)+STR$(T))
770 Z=FNA(A$+"D"+C$+STR$(S)+STR$(T)+" "+STR$(W)+STR$(A))
780 NEXT I
790 L=(((K-N)/2))+N
800 P=R/2
810 X=(P*COS(L))+S
820 Y=(P*SIN(L))+T
830 W=INT(X)
840 A=INT(Y)
850 Z=FNA(A$+"U"+I$+STR$(W)+STR$(A))
860 Z=FNA(A$+"D"+I$+" "+P$+" "+O$+E$)
870 N=K\F=E\GOTO 530
880 RETURN
890 REM
900 P$=CHR$(27)
910 PRINT\PRINT\PRINT\PRINT
920 PRINT "            INSTRUCTIONS FOR USE "
930 PRINT "            -------------------- "\PRINT\PRINT
940 PRINT " Max allowed radius is 3.25 in's or radius's less then"
950 PRINT " 1.00 inches may be used however the increment must be "
960 !     " at least as fine as the number of decimal places inthe radius"
970 !     " For example for radius .375 the increment should be entered "
980 !     " Equal to .001..This is for max resolution--its up to the user"
990 !     " whether ,a more coarse increment is visually satisfactory."
1000 PRINT "    Symbol's used follow the format found in the plotter man."
1010 PRINT " please use it as a reference (Pg.1-20) if unsure."
1020 PRINT "    Circle based polygon's can be generated using a coarse "
1030 PRINT " increment and many rev's(multiples of 2pi). "
1040 INPUT " MORE OR QUIT ( M OR Q )",M$
1050 IF M$="M" THEN 1070
1060 IF M$="Q" THEN 1180
1070 PRINT "    Pie charts and wedge segments are also an option with "
1080 PRINT " the ability to choose different color's per wedge and also "
1090 PRINT " to shade using coarse or fine increments. There is an error "
1100 PRINT " trapping feature that aborts the pie chart sub when you have"
1110 PRINT " exceeded or rather pend on exceeding 100% of a circle. This "
1120 PRINT " feature when it aborts defaults to plot the outer circle so "
1130 PRINT " this can be used to your advantage if your aware of it. Also,"
1140 PRINT " knowing this you should guard against the chance of writing "
1150 PRINT " over the outer circle in an unde sired color."
1160 PRINT "       Well enjoy this routine can be recreational as well as"
1170 PRINT " practical....."\PRINT\PRINT\PRINT
1180 !\!\!\!\!
1190 RETURN
1200 DEF FNA(X$)
1210 X$=X$+CHR$(13)
1220 FOR X=1 TO LEN(X$)
1230 Y=FNO(X$(X,X))
1240 NEXT X
1250 REM
1260 Z = INT(INP(5)/2)\!Z,
1270 IF Z = 2 * INT(Z/2) THEN 1260
1280 X=INP(4)
1290 !"*",X,
1300 RETURN 0
1310 FNEND
1320 DEF FNO(Q$)
```

```
650 Q2=N(I)*(1/(Z4/550))\N4(I)=Q2\NEXT I
660 REM GENERATE X,Y CORD'S FOR GAMMA
670 B=1.579523
680 FOR I = 0 TO 12
690 X=(G4(I)*COS(B))+S\Y=(G4(I)*SIN(B))+T
700 G1(I,1)=INT(X)\G1(I,2)=INT(Y)\B=B-.2617994
710 NEXT I
720 REM GENARATE X,Y CORD'S FOR NEUT.
730 C1=1.57923\FOR I = 0 TO 12
740 X=(N4(I)*COS(C1))+S\Y=(N4(I)*SIN(C1))+T
750 N1(I,1)=INT(X)\N1(I,2)=INT(Y)
760 C1=C1+.2617994\NEXT I
770 REM SECTION TO MARK GAMMA DATA
780 FOR I = 0 TO 12
790 IF G1(I,1)=S AND G1(I,2)=T THEN 940
800 Z=FNA(A$+" "+       "U"+" "+STR$(G1(I,1))+STR$(G1(I,2)))
810 IF C$="NSWC" THEN 850
820 IF C$="NWEF" THEN 870
830 IF C$="LANL" THEN 890
840 IF C$="LLNL" THEN 910
850 Z=FNA(A$+" "+K$+"M33"+" "+STR$(G1(I,1))+STR$(G1(I,2)))
860 Z=FNA(A$+" "+K$+"M31"+" "+STR$(G1(I,1))+STR$(G1(I,2)))\GOTO 940
870 Z=FNA(A$+" "+K$+"M32"+" "+STR$(G1(I,1))+STR$(G1(I,2)))
880 Z=FNA(A$+" "+K$+"M30"+" "+STR$(G1(I,1))+STR$(G1(I,2)))\GOTO 940
890 Z=FNA(A$+" "+K$+"M35"+" "+STR$(G1(I,1))+STR$(G1(I,2)))
900 Z=FNA(A$+" "+K$+"M30"+" "+STR$(G1(I,1))+STR$(G1(I,2)))\GOTO 940
910 Z=FNA(A$+" "+K$+"M33"+" "+STR$(G1(I,1))+STR$(G1(I,2)))
920 Z=FNA(A$+" "+K$+"M22"+" "+STR$(G1(I,1))+STR$(G1(I,2)))
930 Z=FNA(A$+" "+K$+"M21"+" "+STR$(G1(I,1))+STR$(G1(I,2)))
940 NEXT I
950 REM SECTION TO MARK NEUTRON
960 FOR I = 0 TO 12
970 IF N1(I,1)=S AND N1(I,2)=T THEN 1080
980 Z=FNA(A$+" "       +"U"+" "+STR$(N1(I,1))+STR$(N1(I,2)))
990 IF C$="NSWC" THEN 1030
1000 IF C$="NWEF" THEN 1040
1010 IF C$ = "LANL" THEN 1050
1020 IF C$="LLNL" THEN 1060
1030 Z=FNA(A$+" "+P$+"M33"+" "+STR$(N1(I,1))+STR$(N1(I,2)))\GOTO 1080
1040 Z=FNA(A$+" "+P$+"M32"+" "+STR$(N1(I,1))+STR$(N1(I,2)))\GOTO 1080
1050 Z=FNA(A$+" "+P$+"M35"+" "+STR$(N1(I,1))+STR$(N1(I,2)))\GOTO 1080
1060 Z=FNA(A$+" "+P$+"M33"+" "+STR$(N1(I,1))+STR$(N1(I,2)))
1070 Z=FNA(A$+" "+P$+"M22"+" "+STR$(N1(I,1))+STR$(N1(I,2)))
1080 NEXT I
1090 Q3=(3.1415927/Z2)
1100 INPUT "DO YOU WISH TO PLOT THE GAMMA FIELD STRENGTH ( Y OR N )",Y$
1110 IF Y$="N" THEN 1480
1120 REM SUB FOR INTERP AND PLOT -GAMMA
1130 FOR I = 0 TO N-1
1140 X3(I)= G(I)
1150 NEXT I
1160 R2=1.0/(C2+1)\K=0
1170 FOR I = 1 TO N-1\K=K+1
1180 X2(K)=X3(I-1)
1190 X4=(X3(I)-X3(I-1))*R2
1200 J2=1
1210 J2=J2+1
1220 K=K+1
1230 X2(K)=X2(K-1)+X4
1240 IF J2<=C2 THEN 1210
1250 NEXT I\K=K+1
1260 X2(K)=X3(N-1)
1270 FOR I= 1 TO Z2
1280 Q1= X2(I)*(1/(R4/550))\ X2(I)=Q1\NEXT I
1290 B=1.5707963
1300 FOR I = 1 TO Z2
```

PROGRAM LISTING FOR DRAW1

```
10 REM DRAW1 -4TH PART OF POLAR MREM'S PLOTTER
20 REM WRITTEN BY MARSHAL PHIN AUGUST 83
30 E$=CHR$(95)
40 N=13
50 C2=10
60 Z2=((N*C2)+(N-C2))
70 DIM J$(2),N$(2),U$(2),K$(2),R$(3),Z$(3),C$(4),F$(8),G$(8)
80 DIM G1(12,2),N1(12,2),K(N),X(N),B$(2)
90 DIM X2(Z2),Y2(Z2),X3(N),Y3(N),X4(N),Y4(N),X5(Z2),Y5(Z2)
100 DIM G(12),N(12),G2(Z2,2),N2(Z2,2),G4(12),N4(12)
110 DIM A$(80),X$(80),O$(80),Q$(80),T$(80),P$(2),Y$(1)
120 A$=";: IOD 0 A "
130 INPUT "PICK COLOUR FOR GAMMA MARK",K$\!
140 INPUT "PICK COLOUR FOR NEUTRON MARK",P$\!
150 INPUT "PICK COLOUR FOR GAMMA PLOT",J$\!
160 INPUT "PICK COLOUR FOR NEUTRON PLOT",N$\!
170 INPUT " ENTER CODE FOR PARTICULAR LAB'S DATA ",C$
180 INPUT" DID YOU CHOOSE A LIBRARY NAME FOR NEUT. BEFORE(Y OR N)",Y$
190 IF Y$="N" THEN 210
200 INPUT "ENTER NAME OF FILE USED BEFORE ",F$\GOTO 220
210 OPEN#1,"NEUT"\GOTO 230
220 OPEN#1,F$
230 I=0
240 IF TYP(1)=0 THEN 280
250 READ #1 ,N(I)
260 I=I+1
270 GOTO 240
280 CLOSE#1
290 INPUT" DID YOU CHOOSE A LIBRARY NAME FOR GAMM. BEFORE(Y OR N)",Y$
300 IF Y$="N" THEN 310
305 INPUT "ENTER NAME OF FILE USED BEFORE ",G$\GOTO 320
310 OPEN#2, "GAMM"\GOTO 330
320 OPEN #2,G$
330 J=0
340 IF TYP(2)= 0 THEN 380
350 READ #2 ,G(J)
360 J=J+1
370 GOTO 340
380 CLOSE #2
390 OPEN#1 ,"NESC"
400 REM READ NEUTRON SCALE
410 READ#1,Z4
420 CLOSE#1
430 OPEN#2, "GASC"
440 REM READ GAMMA SCALE
450 READ#2 ,R4
460 CLOSE#2
470 S=950\T=700
480 I=.0523599\X=(500*COS(I))+S\Y=(500*SIN(I))+T\W=INT(X)\A=INT(Y)
490 Z=FNA(A$+" "+K$+" "+"U"+STR$(W-50)+STR$(A))
500 Z=FNA(A$+" "+"S12"+" "+STR$(R4)+" "+E$)
510 R2=R4/2\X=(200*COS(I))+S\Y=(200*SIN(I))+T\W=INT(X)\A=INT(Y)
520 Z=FNA(A$+" "+K$+" "+"U"+STR$(W-50)+STR$(A))
530 Z=FNA(A$+" "+"S12"+ STR$(R2)+" "+E$)
540 J=3.0892328\X=(500*COS(J))+S\Y=(500*SIN(J))+T\W=INT(X)\A=INT(Y)
550 Z=FNA(A$+" "+P$+" "+STR$(W-50)+STR$(A))
560 Z=FNA(A$+" "+"S12"+" "+STR$(Z4)+" "+E$)
570 Z3=Z4/2\X=(200*COS(J))+S\Y=(200*SIN(J))+T\W=INT(X)\A=INT(Y)
580 Z=FNA(A$+" "+P$+" "+"U"+" "+STR$(W-50)+STR$(A))
590 Z=FNA(A$+" "+"S12"+" "+STR$(Z3)+" "+E$)
600 REM DATA SCALING AND STORE IN ARRAY\REM GAMMA
610 FOR I = 0 TO 12
620 Q1=G(I) *(1/(R4/550))\G4(I)=Q1\NEXT I
630 REM SCALE NEUTRON
640 FOR I = 0 TO 12
```

```
660 Y=(SQRT((A*A)-(J*J)))*(B/A)
670 I=ABS((Y*200)-T)
680 F=INT(I)
690 Z=FNA(A$+"D"+B$+STR$(G)+STR$(F))
700 NEXT J
710 Z=FNA(A$+" "+B$+" "+"U"+"950,694")
720 Z=FNA(A$+" "+"M34" + E$)
730 Z=FNA(A$+"PO")
740 PRINT " DO YOU WISH TO DRAW POLAR MREM'S OR CALCULATED SOURCE "
750 !\!\INPUT " ENTER 1 FOR POLAR MREM'S; 2 FOR CALC. SOURCE ",Q
760 IF Q= 2  THEN 780
770 CHAIN "DRAW1"\GOTO 790
780 CHAIN "PAL"
790 END
800 DEF FNA(X$)
810 X$=X$+CHR$(13)
820 FOR X=1 TO LEN(X$)
830 Y=FNO(X$(X,X))
840 NEXT X
850 REM
860 Z = INT(INP(5)/2)\!Z,
870 IF Z = 2 * INT(Z/2) THEN 860
880 X=INP(4)
890 !"*",X,
900 RETURN 0
910 FNEND
920 DEF FNO(Q$)
930 Z=INP(5)
940 IF Z/2=INT(Z/2) THEN 930
950 OUT 4,ASC(Q$)
960 !Q$,
970 IF Q$=CHR$(13) THEN !
980 RETURN 0
990 FNEND
```

THE PROGRAM LISTING FOR DRAW

```
10 REM DRAW THIRD PART OF POLAR MREM'S PLOTTER
20 REM WRITTEN BY MARSHAL PHIN AUGUST 83
30 E$=CHR$(95)
40 DIM A$(80),X$(80),O$(80),Q$(80),B$(2),C$(2),D$(2),S$(2),F$(8),G$(8)
50 DIM Y$(1),M$(80),N$(2),J$(2)
60 A$=";: IOD 0 A "
70 INPUT " ENTER MAIN TITLE ",M$\PRINT
80 INPUT " PICK PEN COLOUR FOR MAIN TITLE ",C$\PRINT
90 INPUT "PICK COLOUR FOR GRID",S$\!
100 INPUT "PICK COLOUR FOR LABEL THETA'S",D$\!
110 INPUT " PICK COLOUR FOR NEUT TITLE ",N$\!
120 INPUT " PICK COLOUR FOR GAMM TITLE ",J$\!
130 Z=FNA(A$+C$+" "+"U"+" "+"450,1350"+" "+"U")
140 Z=FNA(A$+" "+C$+" "+"S12"+" "+M$+E$)
150 Z=FNA(A$+N$  +" "+"U"+" "+"250,1200")
160 Z=FNA(A$+N$+" "+"S12"+" "+"NEUTRON"+E$)
170 Z=FNA(A$+J$+" "+"U"+" "+"1550,1200")
180 Z=FNA(A$+J$+" "+"S12"+" "+"GAMMA"+E$)
190 REM DRAW GRID
200 S=950\T=700\Q=1500\P=1200\Z=FNA(A$+S$+" "+"U"+" "+STR$(Q)+STR$(T))
210 FOR I = 0 TO 6.3 STEP .1
220 X=(550*COS(I))+S\Y=(550*SIN(I))+T\W=INT(X)\A=INT(Y)
230 Z=FNA(A$+" "+S$+" "+"D"+" "+STR$(W)+STR$(A))\NEXT I
240 Z=FNA(A$+" "+"U"+S$+" "+STR$(P)+STR$(T))
250 FOR I = 0 TO 6.3 STEP .1
260 X=(250*COS(I))+S\Y=(250*SIN(I))+T\W=INT(X)\A=INT(Y)
270 Z=FNA(A$+" "+"D"+" "+STR$(W)+STR$(A))\NEXT I
280     FOR I = 0 TO 6.283 STEP 1.5707963
290 X=(550*COS(I))+S\Y=(550*SIN(I))+T\W=INT(X)\A=INT(Y)
300 Z=FNA(A$+" "+"U"+S$+" "+STR$(S)+STR$(T))
310 Z=FNA(A$+" "+"D"+S$+" "+STR$(S)+STR$(T)+" "+STR$(W)+STR$(A))
320 NEXT I
330 REM LABEL OF THETA'S
340 Z=FNA(A$+" "+D$+" "+"U"+" "+"1550,1300")
350 B=0\FOR I = 1.5707963 TO  7.5921822 STEP (2*( 1.5707963))
360 X=(600*COS(I))+S\Y=(600*SIN(I))+T\W=INT(X)\A=INT(Y)
370 Z=FNA(A$+" "+D$+" "+"U"+STR$(W)+STR$(A))
380 Z=FNA(A$+" "+"S12"+" "+STR$(B)+" "+E$)
390 B=B+180\NEXT I
400 REM SUB TO DRAW WEAPON
410 A= .063 \ B= .188 \S= 950 \T= 715 \ H = .001
420 B$="P1"
430 Q=(A*200)+S
440 E=INT(Q)
450 Y=(SQRT((A*A)-(A*A)))*(B/A)
460 I=(Y*200)+T
470 F=INT(I)
480 Z=FNA(A$ +"U" +B$ + STR$(E) +STR$(F))
490 FOR J= A TO -A STEP -H
500 P=(J*200)+S
510 G = INT(P)
520 Y=(SQRT((A*A)-(J*J)))*(B/A)
530 I=(Y*200)+T
540 PRINT
550 F=INT(I)
560 Z=FNA(A$+ "D" +B$+ STR$(G) +STR$(F))
570 NEXT J
580 L=(A*200)+S \L1=INT(L)
590 Y=(SQRT((A*A)-(A*A)))*(B/A)
600 I=(Y*200)+T
610 F=INT(I)
620 Z=FNA(A$+"U"+B$+STR$(L1)+STR$(F))
630 FOR J=  A TO -A STEP -H
640 P=ABS((J*200)+S)
650 G=INT(P)
```

THE PROGRAM LISTING FOR SCAL

```
10 REM- SCAL- INTERACTIVE SCALE GENERATION FOR PLOT PURPOSE
20 DIM G(12),N(12),X(12),X1(12),Y$(1),F$(8),G$(8)
30 INPUT" DID YOU CHOOSE A LIBRARY NAME FOR NEUT. BEFORE(Y OR N)",Y$
40 IF Y$="N" THEN 60
50 INPUT "ENTER NAME OF FILE USED BEFORE ",F$\GOTO 70
60 OPEN#1,"NEUT"\GOTO 80
70 OPEN#1,F$
80 I=0
90 IF TYP(1)=0 THEN 130
100 READ #1 ,N(I)
110 I=I+1
120 GOTO 90
130 CLOSE#1
140 INPUT" DID YOU CHOOSE A LIBRARY NAME FOR GAMM. BEFORE(Y OR N)",Y$
150 IF Y$="N" THEN 160
155 INPUT "ENTER NAME OF FILE USED BEFORE ",G$\GOTO 170
160 OPEN#2, "GAMM"\GOTO 180
170 OPEN #2,G$
180 J=0
190 IF TYP(2)= 0 THEN 230
200 READ #2 ,G(J)
210 J=J+1
220 GOTO 190
230 CLOSE #2
240 FOR K = 1 TO I-1
250 IF N(K)<=N(0) THEN 290
260 T1 = N(K)
270 N(K)=N(0)
280 N(0)=T1
290 NEXT K
300 Z4=N(0)
310 FOR I = 1 TO J-1
320 IF G(I)<=G(0) THEN 360
330 T1 = G(I)
340 G(I)=G(0)
350 G(0)=T1
360 NEXT I
370 R4=G(0)
380!\!" THE SCALE FACTOR FOR NEUTRON IS NOW SET AT====",Z4\!\!\!
390INPUT"DO YOU WISH TO CHANGE THE SCALE(Y OR N)",Y$
400 IF Y$="N" THEN 420
410 INPUT"WELL,IF YOU ABSOLUTELY HAVE TOO!--GO AHEAD--INPUT",Z4
420 !\!\!\!"THE SCALE FOR GAMMA IS NOW SET AT====",R4\!\!\!\!
430INPUT"DO YOU WISH TO CHANGE THE SCALE(Y OR N)",Y$
440 IF Y$="N" THEN 460
450 INPUT"OK,GO AHEAD CHANGE THE SCALE===",R4
460 OPEN #4,"GASC"
470 WRITE#4 ,R4
480 CLOSE#4
490 OPEN#5,"NESC"
500 WRITE#5 , Z4
510 CLOSE#5
520 CHAIN "DRAW"
530 END
```

```
640 READ#1,I
650 PRINT I
660 GOTO 620
670 !\!\INPUT "WAS DATA OK (Y OR N )",Y$
680 CLOSE #1
690 IF Y$="N" THEN 200
700 INPUT" DO YOU WISH TO READ A GAMM FILE ( Y OR N )",Y$
710 IF Y$="N"THEN 360
720 INPUT " ENTER FILE TO BE READ ",G$\GOTO 810
730 REM -NEXT FILE FOR GAMMA
740 INPUT " ENTER THE NAME OF YOUR DATA FILE(GAMMA)",G$
750 OPEN #2,G$
760 FOR I=0 TO 12
770 INPUT " ENTER GAMMA DATA ",G
780 WRITE#2,G
790 NEXT I
800 CLOSE#2
810 !\!"NOW LETS CHECK OUR GAMMA DATA STORED ON DISK FILE"\!
820 OPEN #2,G$
830 IF TYP(2)=0 THEN 870
840 READ #2,I
850 PRINT I
860 GOTO 830
870 !\!\INPUT"WAS DATA OK (Y OR N)",Y$
880 CLOSE #2
890 IF Y$="N"THEN 750
900 CHAIN "SCAL"
```

THE PROGRAM LISTING FOR ENTER

```
10 REM--PROGRAM CALLED ENTER-FOR DATA ENTRY
20 DIM Y$(1),F$(8),G$(8)
30 REM- INPUT FOR DATA FILES ASSOCIATED WITH POLAR MREM'S PLOT DRAW & DRAW1
40 REM- WRITTEN BY MARSHAL PHIN SEPT 83
50 INPUT" DO YOU WISH TO CREATE A NEUT FILE ( Y OR N )",Y$\!
60 IF Y$="N" THEN 100
70 INPUT " ENTER NAME OF FILE TO CREATE ",F$\!
80 INPUT " ENTER BLOCK SIZE FOR MEMORY ALLOTTMENT ",B1\!\!
90 CREATE F$,B1
100 INPUT" DO YOU WISH TO CREATE A GAMM FILE ( Y OR N )",Y$\!
110 IF Y$="N" THEN 150
120 INPUT " ENTER NAME OF FILE TO CREATE ",G$\!
130 INPUT " ENTER BLOCK SIZE FOR MEMORY ALLOTTMENT ",B1\!\!
140 CREATE G$,B1
150 INPUT " DO YOU WISH TO READ A NEUT LIBRARY FILE ( Y OR N )",Y$\!
160 IF Y$="N" THEN 180
170 INPUT "ENTER FILE TO BE READ",F$\GOTO 600
180 INPUT " DO YOU WISH TO USE THE DEFAULT FILE OR NAME YOUR OWN(Y OR N)",Y$
190 IF Y$="N" THEN 530
200 OPEN #1, "NEUT"
210 FOR I=0 TO 12
220 INPUT"ENTER NEUTRON DATA",N
230 WRITE#1,N
240 NEXT I
250 CLOSE#1
260 !\!"NOW LETS CHECK OUR NEUTRON DATA STORED ON DISK FILE"\!
270 OPEN#1,"NEUT"
280 IF TYP(1)=0 THEN 330
290 REM ABOVE CHECKS FOR END OF FILE
300 READ#1,I
310 PRINT I
320 GOTO 280
330 !\!\INPUT "WAS DATA OK (Y OR N )",Y$
340 CLOSE #1
350 IF Y$="N" THEN 200
360 INPUT " DO YOU WISH TO USE THE DEFAULT FILE OR NAME YOUR OWN(Y OR N)",Y$
370IF Y$="N" THEN 730
380 OPEN#2 ,"GAMM"
390 FOR I=0 TO 12
400 INPUT " ENTER GAMMA DATA ",G
410 WRITE#2,G
420 NEXT I
430 CLOSE#2
440 !\!"NOW LETS CHECK OUR GAMMA DATA STORED ON DISK FILE"\!
450 OPEN #2,"GAMM"
460 IF TYP(2)=0 THEN 500
470 READ #2,I
480 PRINT I
490 GOTO 460
500 !\!\INPUT"WAS DATA OK (Y OR N)",Y$
510 CLOSE #2
520 IF Y$="N"THEN 370
530 INPUT " ENTER THE NAME OF YOUR DATA FILE ",F$
540 OPEN#1 ,F$
550 FOR I=0 TO 12
560 INPUT"ENTER NEUTRON DATA",N
570 WRITE#1,N
580 NEXT I
590 CLOSE#1
600 !\!"NOW LETS CHECK OUR NEUTRON DATA STORED ON DISK FILE"\!
610 OPEN#1,F$
620 IF TYP(1)=0 THEN 670
630 REM ABOVE CHECKS FOR END OF FILE
```

not want to plot, he must enter an N. The program will then prompt the user to answer the query if he is interested in plotting the neutron field strength. Again, if the user is not interested in plotting, he must enter an N. This allows the user the option of plotting one, both, or none of the fields.

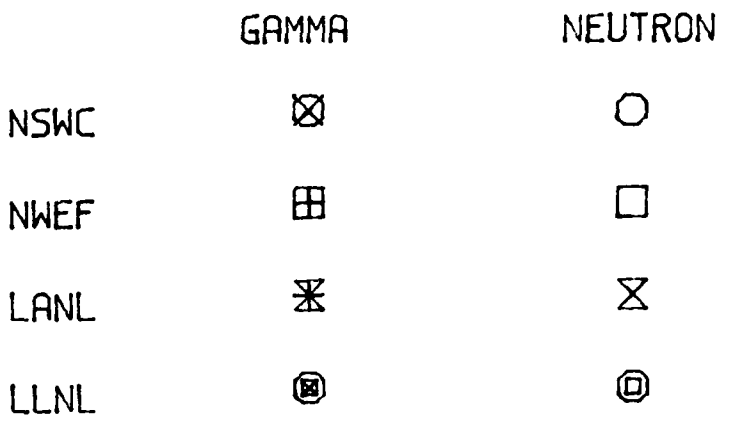


FIGURE 19. THE SYMBOLS USED TO SHOW THE DIFFERENCE BETWEEN LABORATORIES

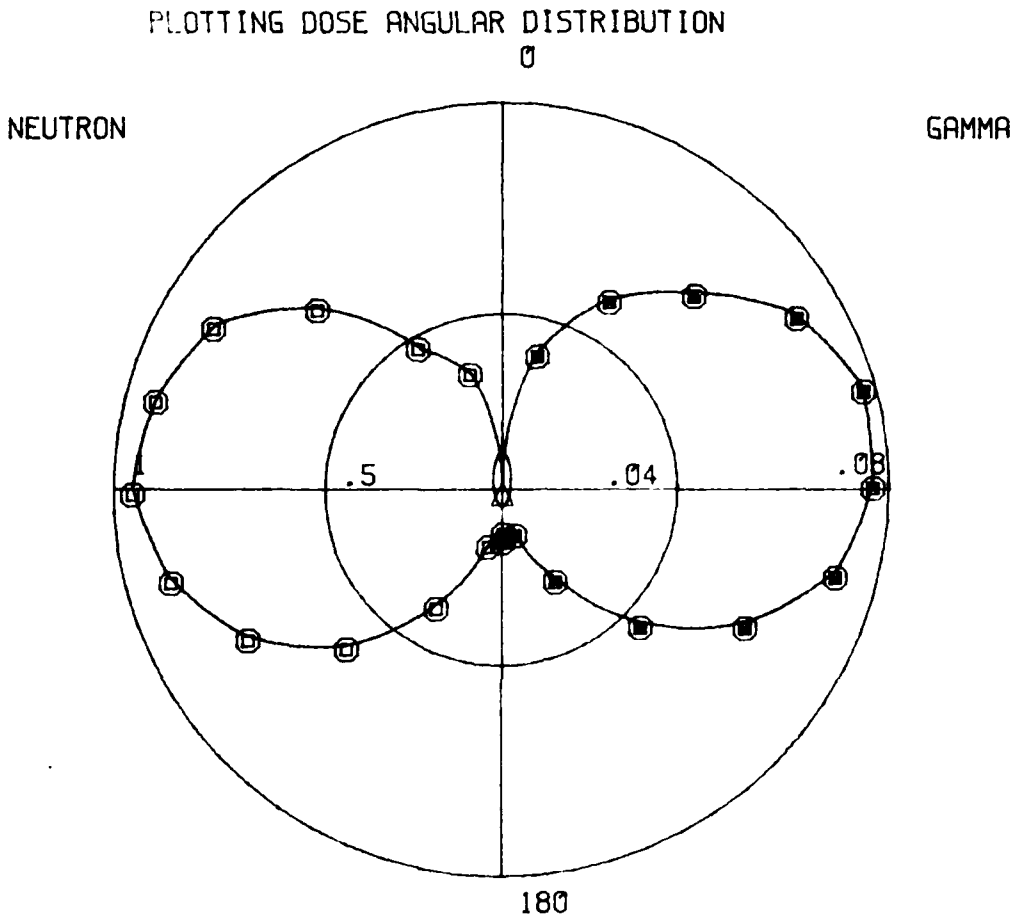


FIGURE 18. THE OUTPUT FROM DRAW1

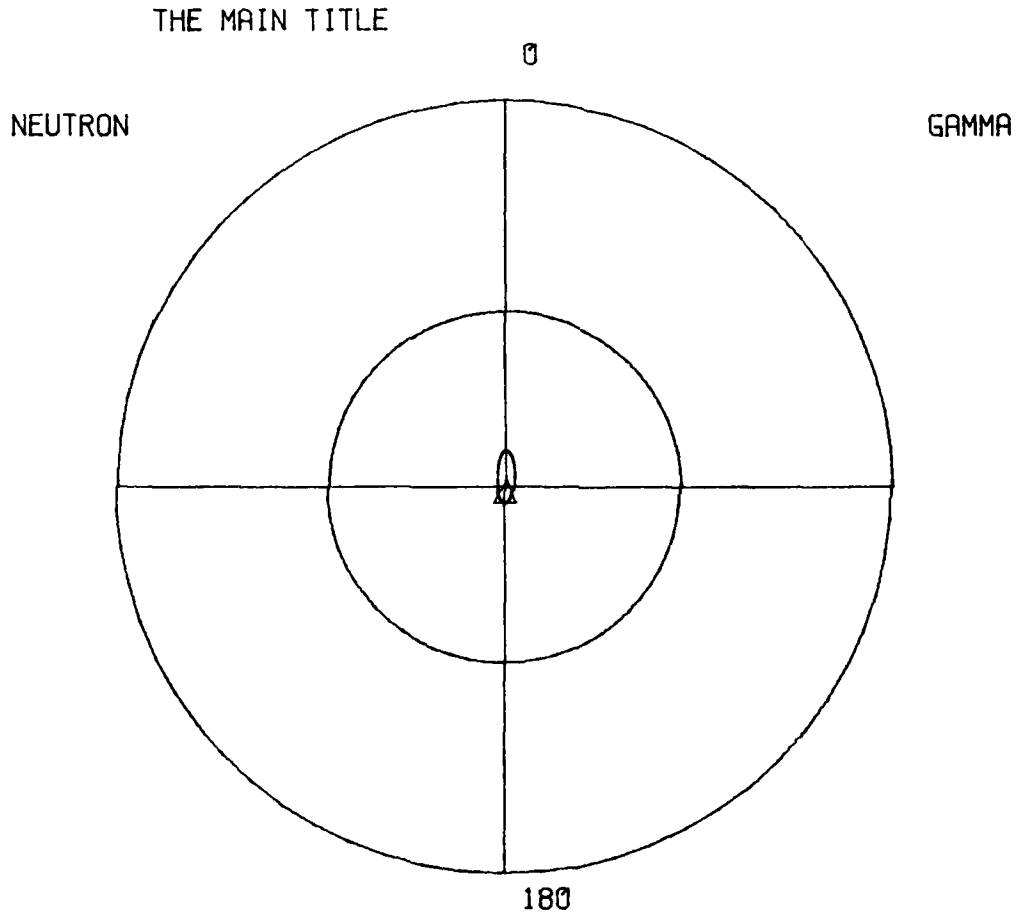


FIGURE 17. THE OUTPUT FROM DRAW

interested in modifying this program to suit there own use. Since this program only draws the polar pattern and does not plot any actual data, this program can be used to draw the grid scale for other polar plotting programs. Draw will ask the user whether he would like to use one or another of two polar plotting programs. This question is asked by the program prompts of line numbers 740 and 750. Figure 17 will show the output from Draw. The program Draw gives the user the option to chain to program Drawl or Pal.

Drawl

This program marks or plots the gamma and neutron data loaded in a file by the program Enter. The user is provided with prompts to enter the color he desires for the gamma data mark, neutron data mark, plot line to show the extent of the gamma field, and plot line to show the extent of the neutron field. This program reads data from files associated with the gamma and neutron data. As in Enter, if the user chooses a file name other then the name used for default, he should answer program lines 180 and 290 with a Y. The two files regarding the scale of the drawing NESC and GASC are read automatically by the program and require no user interaction.

The two concentric circles of the grid shown in Figure 17 describe the limits of full-scale intensity and half-scale intensity, respectively. In the program listing, line numers 470 through 590 are involved with labeling the intensity scale. For example, in Figure 18 the value for full scale on the gamma side of the drawing is .1. This is the value the user chose for full scale in the program Scal and was loaded into the file named GASC. Half the full scale value was marked at the half intensity mark. The I and J values of program lines 480 and 540 describe the angular placement of the scale markings. they are expressed in radians.

The data values for both gamma and neutron are scaled in respect to the actual size of the plotted paper. The line numbers 600 through 620 are responsible for scaling the gamma data and storing this data in a memory array. Similarly line numbers 630 through 650 are responsible for scaling the neutron data and storing the scaled data in another array. This data is scaled in respect to the radius of the largest circle drawn to describe the polar grid. After the data is scaled, the array values for both gamma and neutron are used to create two separate two-dimensional arrays of x and y coordinate pairs. The line numbers associated with this are lines 660 through 760. The data is now ready to be marked.

The original users of this program were concerned with showing the difference between data obtained from four different laboratories. Therefore this program, as listed, is capable of marking the gamma and neutron data in two separate and distinct patterns for each laboratory. The program asks the user to input a four character abbreviation for the laboratory. From this four character code, the program decides which pattern to mark. The symbols used to label differing lab data and the abbreviation for the particular lab are shown in Figure 19.

After the gamma and neutron data are marked the user has the option to plot their field strengths. The first program prompt concerning this will ask the user if he is interested in plotting the gamma field strength. If the user does

neutron data, and GAMM, for gamma data. Please consult Appendix A section regarding files if the terminology used here is not understood. After the data has been entered, Enter will list this data for the user on the terminal screen. This is done to give the user an opportunity to see if the data he has entered was correct. The program provides the user with a prompt asking for an input of Y if the data was correct and an input of N if the data was not correct. If an N is entered, the program will allow the user to reenter the data for the entire file.

Enter can also be used to read data from an already established file. To do this the user should answer Y to the program prompt, would you like to read a file. The user is then given another prompt asking him to input the name of the file in question. The data contained in that file will be listed on the terminal screen. This feature will allow a user to review the data contents of a file and make corrections if desired.

Data entry and management are the sole purposes for Enter. As written, Enter will allow only 13 numbers to be read from or written into a file. Users can modify this easily to suit there own purpose. This would involve changing line numbers 210, 390, 550, and 760 in the program listing. The program Enter will automatically load and run program Scal.

Scal

The purpose of the program Scal is to provide the user with information regarding the comparative magnitude of the data he is planning to plot. Scal looks at all the data in each file used for Gamma and Neutron data respectively. The largest value found in each is displayed on the terminal screen. The user has the choice to use these values as variables used in other programs Draw and Drawl. These variables will be used to scale the drawing and to label the polar grid. A program prompt will ask the user whether he would like to change the scale. A response of Y should be followed by an entry of the scale the user intends to use. A response of N will fix the scale at the value of the data number having the largest magnitude. These variables will be stored in files on the disk called NESC for neutron scale and GASC for gamma scale. The NESC and GASC files will be read by programs Draw and Drawl. The program Scal chains to program Draw.

Draw

Draw is the title of a program module that draws the polar grid and labels the drawing. The user has the opportunity to enter a different main title and choose one of six colors for it. The color of the grid is also chosen by the user. Due to the specific purpose of this program there are two subtitles, one on the upper left side of the page and one on the upper right. These are the labels for the side of the polar grid describing a gamma plot and the side describing a neutron plot respectively. These titles can be edited by changing program lines 160 and 180. As the program stands the user enters a color for each of the titles according to the program prompts. Draw marks two angles on the graph, 0° and 180°. The user can also choose a color for the angles marked. In accordance with its special function, Draw plots a symbol in the center of the grid pattern. This section can, of course, be deleted for those

PLOTTING DOSE ANGULAR DISTRIBUTION

This section deals with a package of four programs that work in sequence to draw a polar plot of experimental data. The four programs are entitled Enter, Scal, Draw, and Drawl. A detailed explanation of each program in the package will follow the brief discussion of their overall purpose as a group.

The objective of the package is to provide a polar plot of the data obtained by measuring the field strength of a radioactive source having an asymmetric dose angular distribution.

An experimenter would obtain the data to be plotted by following this abbreviated procedure. Keeping the distance to the source fixed an experimenter would place an instrument at 13 separate points, each 15° apart from the other. These points then would describe a semicircle with the source at the center. The instrument would record information (number of radioactive particles that collide with the instrument and the time taken to achieve the total count), regarding the radioactive field around the source. There would be two different instruments one for gamma and another for neutron. The above procedure would be done once for each instrument so that a total of 26 data points would be measured. For every one of these measurements for gamma and neutron a term would be calculated called a dose equivalent. It is this term that is plotted in a polar fashion as a magnitude associated with the angle at which it was measured.

FUNCTIONAL BLOCKS

Each program in the package accomplishes a task unique to itself. In this respect they can be thought of as functional blocks. There is a system and order to the interrelationship between the blocks. The most important relationship is that the blocks are chained together. Chaining describes a programming technique where by when one program has finished another is automatically loaded and begins to execute. Various data files are used by the blocks to store information of a global nature. The sequence that the programs are used in is Enter, Scal, Draw, and Drawl.

Enter

Enter is the first program module in the package. This module is used to input fresh data or read data that has previously been saved in a file. All data entered by the user is saved in a file. If the user does not wish to name a specific file to store his data, Enter will use the default files NEUT, for

```
1330 Z=INP(5)
1340 IF Z/2=INT(Z/2) THEN 1330
1350 OUT 4,ASC(Q$)
1360 !Q$,
1370 IF Q$=CHR$(13) THEN !
1380 RETURN 0
1390 REM
1400 FNEND
```

```
1310 X=(X2(I)*COS(B))+S\Y=(X2(I)*SIN(B))+T
1320 IF  X= S AND Y = T THEN 1370
1330 IF  X= 0 AND Y = 0 THEN 1370
1340 H2     =INT(X)\H5     =INT(Y)
1350 Z=FNA(A$+J$+" "+"U"+" "+STR$(H2)+STR$(H5))
1360 GOTO 1390
1370 B=B-Q3
1380 NEXT I
1390 B=1.5707963
1400 FOR I= 1 TO Z2
1410 X=(X2(I)*COS(B))+S\Y=(X2(I)*SIN(B))+T
1420 IF X=0 AND Y=0 THEN 1470
1430 IF X= S AND Y = T THEN 1470
1440 H2     =INT(X)\H5     =INT(Y)
1450 Z=FNA(A$+J$+" "+"D"+" "+STR$(H2)+STR$(H5))
1460 B=B-Q3
1470 NEXT I
1480 INPUT " DO YOU WISH TO PLOT THE NEUTRON FIELD STRENGTH ( Y OR N )",Y$
1490 IF Y$="N" THEN 1850
1500 REM SUB FOR INTERP AND PLOT-
1510 FOR I = 0 TO N-1
1520 X4(I)= N(I)
1530 NEXT I
1540 R2=1.0/(C2+1)\K=0
1550 FOR I = 1 TO N-1\K=K+1
1560 X5(K)=X4(I-1)
1570 X4=(X4(I)-X4(I-1))*R2
1580 J2=1
1590 J2=J2+1
1600 K=K+1
1610 X5(K)=X5(K-1)+X4
1620 IF J2<=C2 THEN 1590
1630 NEXT I\K=K+1
1640 X5(K)=X4(N-1)
1650 FOR I = 1 TO Z2
1660 Q2=X5(I)*(1/(Z4/550))\X5(I)=Q2\NEXT I
1670 C1=1.5707963\FOR I = 1 TO Z2
1680 X=(X5(I)*COS(C1))+S\Y=(X5(I)*SIN(C1))+T
1690 IF X=0 AND Y=0 THEN 1740
1700 IF X=S AND Y=T THEN 1740
1710 H2=INT(X)\H5=INT(Y)
1720 Z=FNA(A$+N$+" "+"U"+" "+STR$(H2)+STR$(H5))
1730 GOTO 1760
1740 C1=C1+Q3
1750 NEXT I
1760 C1=1.5707963
1770 FOR I = 1 TO Z2
1780 X=(X5(I)*COS(C1))+S\Y=(X5(I)*SIN(C1))+T
1790 IF X=0 AND Y=0 THEN 1830
1800 IF X=S AND Y=T THEN 1830
1810 H2     =INT(X)\H5     =INT(Y)
1820 Z=FNA(A$+N$+" "+"D"+" "+STR$(H2)+STR$(H5))
1830 C1=C1+Q3
1840 NEXT I
1850 Z=FNA(A$+"PO")
1860 END
1870 DEF FNA(X$)
1880 X$=X$+CHR$(13)
1890 FOR X=1 TO LEN(X$)
1900 Y=FNO(X$(X,X))
1910 NEXT X
1920 REM
1930 Z  INT(INP(5)/2)\!Z,
1940 IF Z = 2 * INT(Z/2) THEN 1930
1950 X=INP(4)
1960 !"*",X,
```

```
1970 RETURN 0
1980 FNEND
1990 DEF FNO(Q$)
2000 Z=INP(5)
2010 IF Z/2=INT(Z/2) THEN 2000
2020 OUT 4,ASC(Q$)
2030 !Q$,
2040 IF Q$=CHR$(13) THEN !
2050 RETURN 0
2060 FNEND
```

PLOTTING THE NUMERIC APPROXIMATION OF THE DOSE ANGULAR DISTRIBUTION

This section covers a two-program package designed to provide the polar plot of a numeric approximation of the dose angular distribution. The programs that comprise the package are called Pal and Pall. The plot obtained from the package containing Draw and Drawl reveals a picture of the shape of a radioactive field measured by actual experiment. On the other hand, programs Pal and Pall will produce a picture of a numeric approximation to the ideal field.

THE MATHEMATIC BASIS OF THE DOSE APPROXIMATION

This program uses a mathematic basis to plot the numeric approximation of the dose angular distribution. The dose approximation is a polar function.

$$D(R,\theta) = \frac{T(\theta)}{R^2}$$

$$T(\theta) = C\theta + C1 \sin(\theta) + C2 \sin(2\theta) + C3 \sin(4\theta)$$

where

D = the approximated dose

R = the radial distance from the center of the source

T = the trigonometric sine series

θ = the angle

The coefficients of the trigonometric sine series will be different for a specific source. There are also different coefficients depending on the angular displacement. The total range of angular displacement is from 0 to 180°. The user will input four coefficients for the interval from 0 to 45°, four different coefficients for the interval from 45 to 90°, four different values for the interval from 90 to 135°, and four more different values for the interval from 135 to 180°. There must be 16 coefficients entered for each of the gamma and neutron plots.

PAL

The program called Pal performs two functions. The first function allows the user to enter and store data in a file. The program also allows the user to read data already stored in a file. Then the user can accumulate a library of files holding the coefficients of different sources and call on them at any time to produce a plot. The user can name files to store the gamma and neutron coefficients he enters or he can choose to use the default names NECO, for neutron coefficients, and GACO, for gamma coefficients.

The second function provides the user with information regarding the largest magnitude of the does function. This magnitude is shown on the terminal screen and if left unchanged by the user will be used to scale the actual plot. The user is provided with program prompts allowing the opportunity to change the gamma and neutron scale to a value he desires. The gamma scale will be stored in a file named GASC and the neutron scale will be stored in a file named NESC. The program PAL will chain to program PAL1.

PAL1

The program that actually plots the approximation of the angular dose is called Pal1. Pal1 does not draw a polar grid. The user may first use Draw to plot a grid pattern and then proceed with the Pal, Pal1 combination. This is the case shown in Figure 20. The user may also produce a transparency by plotting on mylar with the Pal, Pal1 combination alone. The user may also plot the measured dose angular distribution using the Enter through Draw1 combination, leave the same paper in the plotter and then plot Pal, Pal1 on that same page. The line numbers associated with the plot of the gamma dose approximation are lines 350 through 660 of the program listing. The line numbers associated with the plot of the neutron dose approximation are lines 670 through 970 of the program listing.

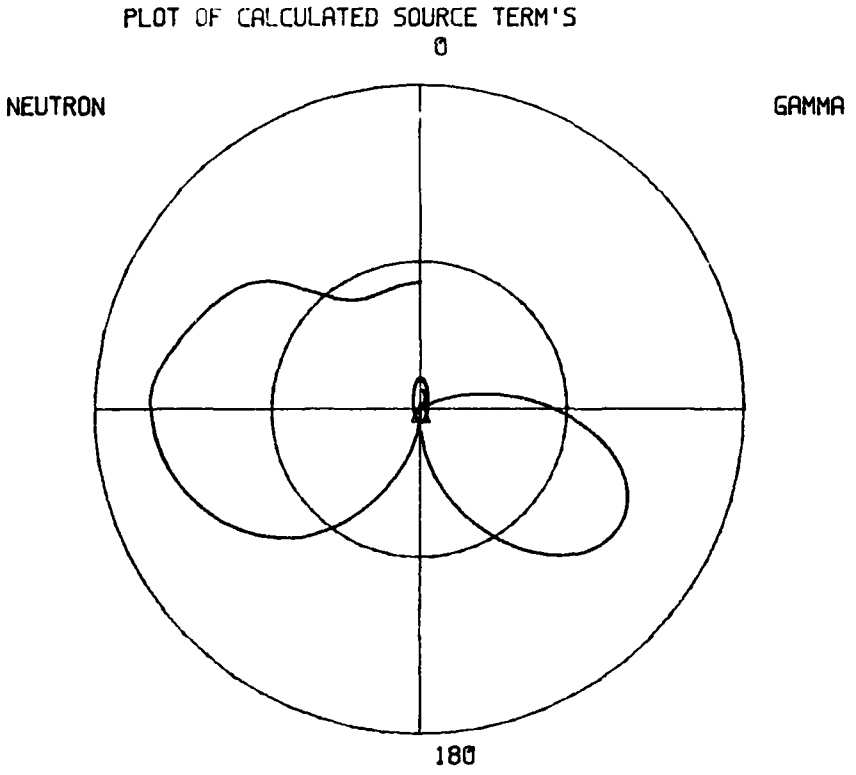


FIGURE 20. THE OUTPUT FROM A DRAW–PAL–PAL1 COMBINATION

THE PROGRAM LISTING FOR PAL

```
10 REM- PAL PROGRM MODULE TO MAKE A DATA FILE OF SOURCE FORMULA
20 REM COEFF. AND A DATA FILE OF SCALE FACTORS
30 REM WRITTEN BY MARSHAL PHIN AUGUST 83
40 DIM N7(15),G7(15),Y$(1),N6(180),G6(180),F$(8),G$(8)
50 N=180
60 INPUT " DO YOU WISH TO CREATE A NEUT COEFF FILE ",Y$\!
70 IF Y$="N" THEN 110
80 INPUT " ENTER THE NAME OF FILE TO CREATE ",F$\!
90 INPUT " ENTER BLOCK SIZE FOR MEMORY ALLOTTMENT ",B1\!\!
100 CREATE F$,B1
110 INPUT " DO YOU WISH TO CREATE A GAMM COEFF. FILE ",Y$\!
120 IF Y$="N" THEN 160
130 INPUT " ENTER THE NAME OF FILE TO CREATE ",G$\!
140 INPUT " ENTER BLOCK SIZE FOR MEMORY ALLOTTMENT ",B1\!\!
150 CREATE G$,B1
160 INPUT " DO YOU WISH TO READ A NEUT COEFF FILE ( Y OR N )",Y$
170 IF Y$="N" THEN 190
180 INPUT " ENTER FILE TO BE READ ",F$\ GOTO 650
190 INPUT " DO YOU WISH TO USE THE DEFAULT FILE OR NAME YOUR OWN(Y OR N)",Y$
200 IF Y$="N" THEN 570
210 OPEN #1, "NECO"
220 !"ENTER COEFFICENTS FOR NEUTRON SOURCE FORMULA"
230 FOR I=0 TO 15
240 INPUT N7(I)
250 WRITE#1,N7(I)
260 NEXT I
270 CLOSE#1
280 !\!"NOW LETS CHECK OUR NEUTRON DATA STORED ON DISK FILE"\!
290 OPEN#1,"NECO"
300 IF TYP(1)=0 THEN 350
310 REM ABOVE CHECKS FOR END OF FILE
320 READ#1,I
330 PRINT I
340 GOTO 300
350 !\!\INPUT "WAS DATA OK (Y OR N )",Y$
360 CLOSE #1
370 IF Y$="N" THEN 210
380 REM -NEXT FILE FOR GAMMA
390 INPUT " DO YOU WISH TO USE THE DEFAULT FILE OR NAME YOUR OWN(Y OR N)",Y$
400 IF Y$="N" THEN 760
410 OPEN#2,"GACO"
420 !"ENTER COEFFICIENT'S FOR GAMMA SOURCE FORMULA "
430 FOR I=0 TO 15
440 INPUT G7(I)
450 WRITE#2,G7(I)
460 NEXT I
470 CLOSE#2
480 !\!"NOW LETS CHECK OUR GAMMA DATA STORED ON DISK FILE"\!
490 OPEN #2,"GACO"
500 IF TYP(2)=0 THEN 540
510 READ #2,I
520 PRINT I
530 GOTO 500
540 !\!\INPUT"WAS DATA OK (Y OR N)",Y$
550 CLOSE #2\GOTO 1400
560 IF Y$="N"THEN 400
570 INPUT " ENTER THE NAME OF YOUR DATA FILE ",F$
580 OPEN#1 ,F$
590 FOR I=0 TO 15
600 INPUT"ENTER NEUTRON DATA",N
610 WRITE#1,N
620 NEXT I
630 CLOSE#1
640 !\!"NOW LETS CHECK OUR NEUTRON DATA STORED ON DISK FILE"\!
```

```
650 OPEN#1,F$
660 I= 0
670 IF TYP(1)=0 THEN 730
680 REM ABOVE CHECKS FOR END OF FILE
690 READ#1,N7(I)
700 PRINT N7(I)
710 I=I+1
720 GOTO 670
730 !\!\INPUT "WAS DATA OK (Y OR N )",Y$
740 CLOSE #1
750 IF Y$="N" THEN 210
760 REM -NEXT FILE FOR GAMMA
770 INPUT " DO YOU WISH TO READ A GAMM COEFF FILE ( Y OR N )",Y$
780 IF Y$="N" THEN 390
790 INPUT " ENTER FILE TO BE READ ",G$\GOTO 880
800 INPUT " ENTER THE NAME OF YOUR DATA FILE(GAMMA)",G$
810 OPEN #2,G$
820 FOR I=0 TO 15
830 INPUT " ENTER GAMMA DATA ",G
840 WRITE#2,G
850 NEXT I
860 CLOSE#2
870 !\!"NOW LETS CHECK OUR GAMMA DATA STORED ON DISK FILE"\!
880 OPEN #2,G$
890 I = 0
900 IF TYP(2)=0 THEN 950
910 READ #2,G7(I)
920 PRINT G7(I)
930 I=I+1
940 GOTO 900
950 !\!\INPUT"WAS DATA OK (Y OR N)",Y$
960 CLOSE #2
970 IF Y$="N"THEN 810
980 INPUT " ENTER DESIRED RADIAL DISTANCE FOR SOURCE FORMULA'S",D3
990 Q4=(3.1415927/180)
1000 B= 1.5707963\B4=B-1.5707963
1010 J=0
1020 REM LOOP-BACK RETURN
1030 IF B>= 1.5707963 AND B< 2.3561945 THEN 1070
1040 IF B>= 2.3561945 AND B  < 3.14115926 THEN 1100
1050 IF B>= 3.1415926 AND B  < 3.9269908 THEN 1130
1060 IF B>= 3.9269908 AND B < 4.712389 THEN 1160
1070 B4=B-1.5707963
1080 R9=(G7(0)+(G7(1)*SIN(B4))+(G7(2)*SIN(2*B4))+(G7(3)*SIN(4*B4)))
1090 GOTO 1180
1100 B4=B-1.5707963
1110 R9=(G7(4)+(G7(5)*SIN(B4))+(G7(6)*SIN(2*B4))+(G7(7)*SIN(4*B4)))
1120 GOTO 1180
1130 B4=B-1.5707963
1140 R9=(G7(8)+(G7(9)*SIN(B4))+(G7(10)*SIN(2*B4))+(G7(11)*SIN(4*B4)))
1150 GOTO 1180
1160 B4=B-1.5707963
1170 R9=(G7(12)+(G7(13)*SIN(B4))+(G7(14)*SIN(2*B4))+(G7(15)*SIN(4*B4)))
1180 T9=R9/(D3*D3)
1190 G6(J)=T9
1200 J=J+1
1210 B=B-Q4
1220 IF B>= -1.5707964 THEN 1020
1230 K1=J
1240 REM START PLOT NEUTRON SOURCE FIELD THEORITICIAL
1250 Q4=(3.1415927/180)
1260 B= 1.5707963
1270 B4=B-1.5707963
1280 J=0
1290 REM LOOP-BACK RETURN
1300 IF B>= 1.5707963 AND B  < 2.3561945 THEN 1340
```

```
1310 IF B>= 2.3561945 AND  B < 3.14115926 THEN 1370
1320 IF B>= 3.1415926 AND B  < 3.9269908 THEN 1400
1330 IF B>= 3.9269908 AND B  < 4.712389 THEN 1430
1340 B4=B-1.5707963
1350 R9=(N7(0)+(N7(1)*SIN(B4))+(N7(2)*SIN(2*B4))+(N7(3)*SIN(4*B4)))
1360 GOTO 1450
1370 B4=B-1.5707963
1380 R9=(N7(4)+(N7(5)*SIN(B4))+(N7(6)*SIN(2*B4))+(N7(7)*SIN(4*B4)))
1390 GOTO 1450
1400 B4=B-1.5707963
1410 R9=(N7(8)+(N7(9)*SIN(B4))+(N7(10)*SIN(2*B4))+(N7(11)*SIN(4*B4)))
1420 GOTO 1450
1430 B4=B-1.5707963
1440 R9=(N7(12)+(N7(13)*SIN(B4))+(N7(14)*SIN(2*B4))+(N7(15)*SIN(4*B4)))
1450 T9=R9/(D3*D3)
1460 N6(J)=T9
1470 J=J+1
1480 B=B+Q4
1490 IF B<= 4.712389 THEN 1290
1500 FOR K = 1 TO N-1
1510 IF N6(K)<=N6(0) THEN 1550
1520 T1 = N6(K)
1530 N6(K)=N6(0)
1540 N6(0)=T1
1550 NEXT K
1560 Z4=N6(0)
1570 FOR I = 1 TO N-1
1580 IF G6(I)<=G6(0) THEN 1620
1590 T1 = G6(I)
1600 G6(I)=G6(0)
1610 G6(0)=T1
1620 NEXT I
1630 R4=G6(0)
1640 !\!" THE SCALE FACTOR FOR NEUTRON IS NOW SET AT====",Z4\!\!\!
1650 INPUT"DO YOU WISH TO CHANGE THE SCALE(Y OR N)",Y$
1660 IF Y$="N" THEN 1680
1670 INPUT"WELL,IF YOU ABSOLUTELY HAVE TOO!--GO AHEAD--INPUT",Z4
1680 !\!\!\!"THE SCALE FOR GAMMA IS NOW SET AT====",R4\!\!\!\!
1690 INPUT"DO YOU WISH TO CHANGE THE SCALE(Y OR N)",Y$
1700 IF Y$="N" THEN 1720
1710 INPUT"OK,GO AHEAD CHANGE THE SCALE===",R4
1720 OPEN #4,"GASC"
1730 WRITE#4 ,R4
1740 CLOSE#4
1750 OPEN#5,"NESC"
1760 WRITE#5 , Z4
1770 CLOSE#5
1780 CHAIN "PAL1"
1790 END
```

THE PROGRAM LISTING FOR PAL1

```
10 REM --PAL1  MODIFIED PAL TO PLOT SOURCE FORMULA
20 REM WRITTEN BY MARSHAL PHIN AUGUST 83
30 E$=CHR$(95)
40 DIM N7(15),G7(15),J$(2),N$(2),A$(80),X$(80),Q$(80),Y$(1),F$(8),G$(8)
50 A$=";: IOD 0 A "
60 INPUT "PICK COLOUR FOR GAMMA PLOT",J$\!
70 INPUT "PICK COLOUR FOR NEUTRON PLOT",N$\!
80 INPUT " DID YOU CHOOSE A LIBRARY FILE NAME FOR NEUT BEFORE(Y OR N)",Y$
90 IF Y$="N" THEN 110
100 INPUT " ENTER NAME OF FILE USED BEFORE ",F$\GOTO 120
110 OPEN#1,"NECO"\GOTO 130
120 OPEN#1 ,F$
130 I=0
140 IF TYP(1)=0 THEN 170
150 READ #1,N7(I)
160 I=I+1\GOTO 140
170 CLOSE#1
180 INPUT " DID YOU CHOOSE A LIBRARY FILE NAME FOR GAMM BEFORE(Y OR N)",Y$
190 IF Y$="N" THEN 210
200 INPUT " ENTER NAME OF FILE USED BEFORE ",G$\GOTO 220
210 OPEN#2 , "GACO"\GOTO 230
220 OPEN#2 ,G$
230 I=0
240 IF TYP(2)=0 THEN 270
250 READ#2, G7(I)
260 I=I+1\GOTO 240
270 CLOSE #2
280 OPEN#3 , "GASC"
290 READ #3,R4
300 CLOSE#3
310 OPEN #4, "NESC"
320 READ #4,Z4
330 CLOSE#4
340 INPUT " ENTER DESIRED RADIAL DISTANCE FOR SOURCE FORMULA'S",D3
350 S=950\T=700
360 REM START PLOT GAMMA THEORITICAL
370 Q4=(3.1415927/180)
380 B= 1.5707963\B4=B-1.5707963
390 R9=(G7(0)+(G7(1)*SIN(B4))+(G7(2)*SIN(2*B4))+(G7(3)*SIN(4*B4)))
400 PRINT" R9 EQUALS",R9
410 T9 = (R9/(D3*D3))\T8=T9*(1/(R4/500))
420 X=(T8*COS(B))+S\Y=(T8*SIN(B))+T
430 H2      =INT(X)\H5      =INT(Y)
440 Z=FNA(A$+J$+" "+"U"+" "+STR$(H2)+STR$(H5))
450 REM LOOP-BACK RETURN
460 IF B>= 1.5707963 AND B< 2.3561945 THEN 500
470 IF B>= 2.3561945 AND B  < 3.14115926 THEN 530
480 IF B>= 3.1415926 AND B  < 3.9269908 THEN 560
490 IF B>= 3.9269908 AND B < 4.712389 THEN 590
500 B4=B-1.5707963
510 R9=(G7(0)+(G7(1)*SIN(B4))+(G7(2)*SIN(2*B4))+(G7(3)*SIN(4*B4)))
520 GOTO 610
530 B4=B-1.5707963
540 R9=(G7(4)+(G7(5)*SIN(B4))+(G7(6)*SIN(2*B4))+(G7(7)*SIN(4*B4)))
550 GOTO 610
560 B4=B-1.5707963
570 R9=(G7(8)+(G7(9)*SIN(B4))+(G7(10)*SIN(2*B4))+(G7(11)*SIN(4*B4)))
580 GOTO 610
590 B4=B-1.5707963
600 R9=(G7(12)+(G7(13)*SIN(B4))+(G7(14)*SIN(2*B4))+(G7(15)*SIN(4*B4)))
610 T9=R9/(D3*D3)
620 T8=T9*(1/(R4/500))\X=(T8*COS(B))+S  \Y=(T8*SIN(B))+T
630 H2      =INT(X)\H5      =INT(Y)
640 Z=FNA(A$+J$+" "+"D"+" "+STR$(H2)+STR$(H5))
```

```
650 B=B-Q4
660 IF B>= -1.5707964 THEN 450
670 REM START PLOT NEUTRON SOURCE FIELD THEORITICIAL
680 Q4=(3.1415927/180)
690 B4=B-1.5707963
700 B= 1.5707963
710 R9=(N7(0)+(N7(1)*SIN(B4))+(N7(2)*SIN(2*B4))+(N7(3)*SIN(4*B4)))
720 T9 = (R9/(D3*D3))\T8=T9*(1/(Z4/500))
730 X=(T8*COS(B))+S\Y=(T8*SIN(B))+T
740 H2      =INT(X)\H5      =INT(Y)
750 Z=FNA(A$+N$+" "+"U"+" "+STR$(H2)+STR$(H5))
760 REM LOOP-BACK RETURN
770 IF B>= 1.5707963 AND B  < 2.3561945 THEN 810
780 IF B>= 2.3561945 AND  B < 3.14115926 THEN 840
790 IF B>= 3.1415926 AND B  < 3.9269908 THEN 870
800 IF B>= 3.9269908 AND B  < 4.712389 THEN 900
810 B4=B-1.5707963
820 R9=(N7(0)+(N7(1)*SIN(B4))+(N7(2)*SIN(2*B4))+(N7(3)*SIN(4*B4)))\! R9
830 GOTO 920
840 B4=B-1.5707963
850 R9=(N7(4)+(N7(5)*SIN(B4))+(N7(6)*SIN(2*B4))+(N7(7)*SIN(4*B4)))
860 GOTO 920
870 B4=B-1.5707963
880 R9=(N7(8)+(N7(9)*SIN(B4))+(N7(10)*SIN(2*B4))+(N7(11)*SIN(4*B4)))
890 GOTO 920
900 B4=B-1.5707963
910 R9=(N7(12)+(N7(13)*SIN(B4))+(N7(14)*SIN(2*B4))+(N7(15)*SIN(4*B4)))
920 T9=R9/(D3*D3)
930 T8=T9*(1/(Z4/500))\X=(T8*COS(B))+S  \Y=(T8*SIN(B))+T
940 H2      =INT(X)\H5      =INT(Y)
950 Z=FNA(A$+N$+" "+"D"+" "+STR$(H2)+STR$(H5))
960 B=B+Q4
970 IF B<= 4.712389 THEN 760
980 Z=FNA(A$+"PO")
990 END
1000 DEF FNA(X$)
1010 X$=X$+CHR$(13)
1020 FOR X=1 TO LEN(X$)
1030 Y=FNO(X$(X,X))
1040 NEXT X
1050 REM
1060 Z = INT(INP(5)/2)\!Z,
1070 IF Z = 2 * INT(Z/2) THEN 1060
1080 X=INP(4)
1090 !"*",X,
1100 RETURN 0
1110 FNEND
1120 DEF FNO(Q$)
1130 Z=INP(5)
1140 IF Z/2=INT(Z/2) THEN 1130
1150 OUT 4,ASC(Q$)
1160 !Q$,
1170 IF Q$=CHR$(13) THEN !
1180 RETURN 0
1190 FNEND
```

BIBLIOGRAPHY

Hiplot Digital Plotter Operators Instructions, Houston Instrument Division of Bausch and Lomb.

System Software Manual, Revision 2.1, Northstar Computers Inc., pp. A-1 to M-19.

APPENDIX A

DISK FILE MANAGEMENT

This appendix is intended to be a guide to disk file management under the Northstar Basic program language. It is not to be used as a universal guide. If the programs in this document are to be adopted for use with another language the person involved should make every effort to learn the particular methodology and command structure for the language in question.

Data is stored on a diskette in FILES. A file is a section of storage space on the diskette which is reserved for data storage use by giving it a FILE NAME. The name of a file consists of a series of not more than eight printable characters. The printable characters include the upper and lower case alphabets, the digits 0 to 9, and the various punctuation symbols excluding the use of the space and the comma.

An additional attribute given to a file is the size. The size of a file is specified in FILE BLOCKS. A file block is 256 bytes of information. In a double-density system, each file must have an even number of file blocks because file space on diskette is allocated in terms of SECTORS. Two file blocks will fit in one sector of a double-density system. In single-density systems, a disk block is the same as a sector.

Before a file may be used to store data it must be created. This means the disk must reserve space for the file. The CREATE statement may be used to create a file of any size. The programs in this document prompt the user to enter a name for a file and the size of the file. The user can approximate the file size by remembering each digit of his numeric data will occupy one byte in memory. Once created, the file's size in file blocks is fixed. The amount of information in that file can never exceed the allocated space.

Many file handling activities are handled automatically in the programs in this document. A discussion has been made of only the elements of file usage the user will actively take part in. If more detailed information is needed consult the System Software Manual for the Northstar Computer.

DISTRIBUTION

	Copies
avy Regional Data Automation Center ttn: Code 32 (L. Slater) orfolk, VA 22217	1
avy Regional Data Automation Center ttn: Code 424 (M. Miller) uilding 143 ashington Navy Yard ashington, DC 20374	1
allistic Research Laboratory ttn: DRSMC-BLI(A) (G. Keller) berdeen Proving Ground, MD 21005	1
efense Technical Information Center ameron Station lexandria, VA 22314	12
ibrary of Congress ttn: *Gift and Exchange* Division ashington, DC 20540	4
Micrographic Users Group P.O. Box 15720 Austin, TX 78761	1
Internal Distribution:	
K34 (T. Orlow)	1
K52 (K. Pitts)	1
K301 (H. Thombs)	1
N14 (R. Hewitt)	1
R34 (R. Lee)	1
R42 (G. Goo)	1
R41 (G. K. Riel)	5
R41 (N. Rao)	1
R41 (R. Spurlock)	1
R41 (L. Murray)	1
R41 (P. Winters)	1
R41 (G. Levine)	1
R41 (J. M. Phin)	6
X33 (W. Cobbin)	1
X33 (S. Phin)	6
X33 (A. Douyon)	1
X33 (R. Whitman)	1
E431	9
E432	3
E35	1

END

FILMED

7-85

DTIC